AVE Trends in Intelligent Informatics Reports

AI-Based Lung Disease Prediction Using Machine Learning with PCA-Based Dimensionality Reduction

Nikhil Baiju Punnen[1], M. Pranav[2], Allent S. Manakatt[3], R. Regin[4,*], K. Senthamilselvan[5], S. Tejas[6]

[1,2,3,4]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[5]Department of Electronics and Communication Engineering, Dhaanish Ahmed College of Engineering, Chennai, Tamil Nadu, India.
[6]Department of Data Science, Analytics and Engineering, Arizona State University, Tempe, Arizona, United States of America.
nb5519@srmist.edu.in[1], pm7703@srmist.edu.in[2], as3976@srmist.edu.in[3], regin12006@yahoo.co.in[4], senthamilkselva@gmail.com[5], tsundar1@asu.edu[6]

*Corresponding author

Abstract: The objective of this paper is to conduct a thorough comparative study of baseline and preprocessed (with PCA, a technique for dimensionality reduction) classical machine learning classifiers: Logistic Regression (LR), Support Vector Machine (SVM), and Random Forest (RF) for lung disease prediction using high-dimensional biomedical data. Lung diseases (pneumonia, tuberculosis, COPD, lung cancer) pose a tremendous global health challenge and are responsible for several million deaths per year. There is an urgent need for prompt, accurate and automated detection of lung diseases to effectively improve patient care and reduce the variability inherent in human interpretation of diagnostic results. Machine learning with high-dimensional biomedical data is characterized by difficulties related to feature interdependencies, redundancy, correlation, and noise amplification, all of which increase the risk of overfitting, particularly when the number of training samples is much smaller than the number of features. This study compares six classification pipelines (three baselines, three preprocessed with PCA) across three train-test partition ratios (80/20, 70/30, 60/40) and two cross-validation methods (3-fold and 5-fold) using a leakage-free nested cross-validation methodology that computed feature scaling and PCA transformations only from the training folds. Our experiments show that baseline classifiers achieve training accuracies of 97–100% and significantly lower test accuracies, indicating overfitting in the high-dimensional domain. In contrast, PCA pipelines preprocess and improve generalization by narrowing the train-test gap across all three ratios.

Keywords: Machine Learning; Dimensionality Reduction; Support Vector Machine; Random Forest; Logistic Regression; Biomedical Data; Lung Disease Prediction; Classification Pipelines; Nested Cross-Validation.

Cite as: N. B. Punnen, M. Pranav, A. S. Manakatt, R. Regin, K. Senthamilselvan, and S. Tejas, "AI-Based Lung Disease Prediction Using Machine Learning with PCA-Based Dimensionality Reduction," *AVE Trends in Intelligent Informatics Reports*, vol. 1, no. 1, pp. 52–66, 2026.

Journal Homepage: https://www.avepubs.com/user/journals/details/ATIIR

Received on: 01/05/2025, **Revised on:** 16/06/2025, **Accepted on:** 15/08/2025, **Published on:** 01/03/2026

DOI: https://doi.org/10.64091/ATIIR.2026.000280

1. Introduction

One of the major threats to public health worldwide, diseases of the lungs are a group of conditions responsible for a wide array of morbidity and mortality. From the World Health Organization's information, lower respiratory infections, Chronic Obstructive Pulmonary Disease and lung carcinoma constitute 3 of the 10 leading causes of global death, contributing millions of premature deaths annually [1]. Pneumonia, tuberculosis, chronic obstructive pulmonary disease, and primary lung carcinoma represent a diverse group of pathologies [20]. They vary significantly in their causes, clinical courses, and treatment requirements; however, all share one fundamental requirement – early and precise diagnosis [21]. Traditional approaches to diagnosis rely mostly on physical examination, Pulmonary Function Tests and radiological methods (CXR, CT scan) [22]. However, these approaches require time, require expertise to provide accurate information, and are subjective and prone to inter-observer variability, especially in resource-limited environments [23]. Integrating machine learning (ML) into clinical diagnosis will provide an objective method for acquiring large amounts of biomedical data, discovering consistent diagnostic markers of disease, supporting clinical decision-making, and providing confidence estimates of disease presence using popular ML classifiers [2]. Most biomedical datasets, whether obtained through gene expression analysis, clinical feature extraction, or image processing, tend to have dozens to thousands of features, often exceeding the number of samples that can be feasibly acquired [24]; [30].

Such "curse of dimensionality" results in feature redundancy, intercorrelated noise, and increased computational burden, predisposing classifiers to overfitting on the training set and to poor generalization on the test set [25]; [26]. PCA offers a well-characterized linear method of reducing high-dimensional data to a low-dimensional orthogonal projection space, spanned by a set of uncorrelated eigenvectors that capture most of the data variance [5]; [27]. This paper attempts to answer three core research questions:

- How do three baseline classifiers, LR, SVM and RF, perform on a lung disease dataset with high feature dimensionality in accuracy, macro-F1 and ROC-AUC?
- Is there evidence of enhanced generalization performance, in terms of reduced overfitting, from PCA-based classifier preprocessing across the multiple data splitting schemes?
- Which classifier–preprocessing combination demonstrates the maximum number of consistently high-performing trials across multiple experimental configurations?

This paper conducts a rigorous, multi-run evaluation of multiple classification pipelines across multiple split ratios and cross-validation methodologies, under an exacting, leakage-free research protocol, to produce reproducible, comparative results that inform clinical applications of PCA-enhanced ML classifiers for lung disease prediction [28]; [29].

2. Literature Review

2.1. Machine Learning in Lung Disease Diagnosis

Research on machine learning for the diagnosis of lung disease has been widely studied for over 20 years. Sarker [3] provides a comprehensive review of machine learning classifiers in biomedical diagnostic systems, outlining the variety of ML techniques used in biomedical applications. Support Vector Machine, Random Forest and Logistic Regression appear to be consistently strong performers on clinical datasets. A survey on machine learning classifiers for cancer categorization using gene expression data by Alharbi and Vakanski [6] recommends that strong classifier performance on complex, high-dimensional biomedical problems must be achieved through prior feature space reduction, as machine learning classifiers on raw data were consistently over-fitted. Pellegrino et al. [7], when applying an RF classifier to variant prediction for abnormality detection in clinical genome analysis in cancer research, demonstrated the efficiency of this method for handling imbalanced, high-dimensional data via ensemble voting. Kallah-Dagadu et al. [8], by applying interpretable ML classifiers and feature selection for breast cancer prediction, offer state-of-the-art accuracy and clinical interpretability. Nyakundi et al. [9] are the only ones to address imbalance in gene expression datasets using cost-sensitive learning and have demonstrated that accuracy can be improved.

2.2. Dimensionality Reduction and PCA

PCA is the most widely studied dimensionality reduction method in machine learning for biomedical applications. The established formal theory for PCA, relating the maximized variance to the eigenvectors of the covariance matrix, is due to Jolliffe and Cadima [5], which identifies the major modes of variance with the eigenvectors corresponding to the largest eigenvalues, while noise-dominated modes are discarded. Hasan and Abdulazeez [4] comprehensively evaluate PCA against alternative dimension reduction methods; kernel PCA, independent component analysis and t-SNE and found linear PCA to have the optimal balance of information preservation and computational efficiency in a supervised classification task with a moderate number of samples. An investigation focusing solely on the effect of dimensionality reduction methods on the performance of ML classifiers for cancer prediction was conducted by Kabir et al. [11]. They concluded that all ML classifiers

used (SVM and logistic Regression) benefited significantly from PCA preprocessing, and that test-set generalization performance increased in all cases. They argue this is a result of reduced noise and irrelevant, multicollinear variables. New recommendations concerning "best practice" PCA for science have been put forward by Greenacre et al. [12], stating that cumulative explained variance should be the ultimate test and that PCA should not be fitted to more than the training data to avoid leakage into the test set Wang et al. [13].

2.3. Class Imbalance and Evaluation Frameworks

Class imbalance is a common occurrence in clinical data, where disease is often substantially under-represented when compared to a large set of healthy controls. The He and Garcia [14] paper is the classic on learning with imbalanced data. It divides solutions into data-level resampling techniques, algorithm-level cost-sensitive classifiers, and hybrid resampling/feature selection methods. They show that accuracy is an unreliable measure of performance on imbalanced data and argue that the macro-averaged F1-score and ROC area under the curve are the main evaluation criteria. Bommert et al. [15] compare the effectiveness of several filter-based feature selection algorithms on high-dimensional survival data, demonstrating that data preprocessing, including feature selection and dimensionality reduction, is key to accurate classification. Ghaleb et al. [10] showed that, in very imbalanced classification problems, SMOTE oversampling outperforms non-resampling baselines, such as Random Forest, with a reported macro-F1 up to 15% higher. Sugianto and Wahyuningsih [16] experimented with an SMOTE-based augmentation of RF-based vehicle classifiers, demonstrating the approach's generalisability to new domains. Both works support the approach taken here, which uses stratified sampling and class-balanced metrics.

2.4. Research Gap and Novelty

This work was conducted within a comparative experimental approach based on supervised binary and multi-class classifiers classification paradigms. The experimental design aimed to assess, separately and together, the contributions of three classical machine learning classifiers and the Principal Component Analysis (PCA) reduction method to the task of predicting lung diseases. For this purpose, six different classification pipelines were proposed. The three existing classifiers were trained once on the raw, preprocessed and normalized set of features, and three others were trained on the normalized set of features transformed by PCA-based reduction.

3. Methodology

3.1. Research Design

The study employs a comparative experimental setup based on a supervised binary and multi-class classification scheme. The experimental design focuses on assessing the independent and synergistic effects of three classical ML classifiers and PCA-based feature extraction on lung disease classification performance.

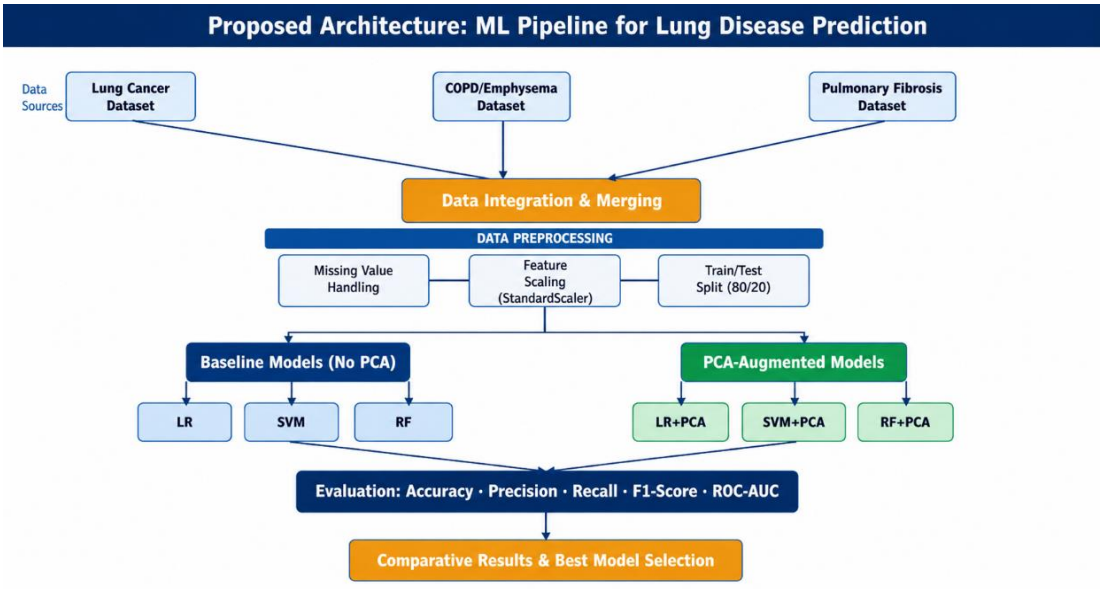


Figure 1: Illustrates the end-to-end architecture of the proposed machine learning pipeline

Six classification pipelines are proposed: three standard classifiers applied directly to the raw, standardized feature space (LR, SVM and RF), and three classification models based on PCA feature extraction as a mid-stage between feature scaling and classification (LR+PCA, SVM+PCA and RF+PCA). All classification models are assessed under identical experimental conditions to enable a fair comparison. Figure 1 presents the end-to-end architecture of the proposed machine learning pipeline. For each experiment, the raw data from 3 lung disease datasets (downloaded from open sources) are aggregated and preprocessed (imputation, StandardScaler and an 80/20 train/test split). The resulting features are then placed on six classification pipelines (3 baseline models, LR, SVM, RF - on the original feature space, and 3 PCA versions, PCA as the intermediate step between the original feature space and the classifier) and evaluated using the same experimental setup and classifying metrics (Accuracy, Precision, Recall, F1-Score, ROC-AUC) in an unbiased way. The overall performance on 3 datasets then determines the best model.

3.2. Algorithm

The proposed methodology is formalized in Algorithm 1, which outlines the end-to-end pipeline for lung disease prediction using classical machine learning classifiers with and without PCA-based dimensionality reduction**.**

Algorithm 1: Lung Disease Prediction via ML Classifiers with and without PCA:

- **Input:** Raw lung disease datasets D = {D1, D2, D3}; classifiers C = {LR, SVM, RF}; PCA variance threshold θ = 0.95.
- **Output:** Performance metrics {Accuracy, Precision, Recall, F1-Score, ROC-AUC} for each pipeline; optimal model selection.

Phase 1: Data Preprocessing

1: for each dataset Di in D do
2: Remove features with near-zero variance ($\sigma^2 < 0.001$); apply winsorization for outliers
3: Standardize features using Z-score normalization: $z = (x - \mu) / \sigma$
4: Split Di into training set Dtrain (80%) and test set Dtest (20%) using stratified sampling
5: end for

Phase 2: Baseline Pipeline (Without PCA)

6: for each classifier Cj in {LR, SVM, RF} do
7: Train Cj on Dtrain; predict labels on Dtest
8: Compute and record {Accuracy, Precision, Recall, F1-Score, ROC-AUC} for pipeline Cj
9: end for

Phase 3: PCA-Augmented Pipeline

10: for each classifier Cj in {LR+PCA, SVM+PCA, RF+PCA} do
11: Fit PCA on Dtrain; retain components explaining θ = 95% of cumulative variance
12: Transform Dtrain and Dtest using fitted PCA projection matrix
13: Train Cj on PCA-reduced Dtrain; predict labels on PCA-reduced Dtest
14: Compute and record {Accuracy, Precision, Recall, F1-Score, ROC-AUC} for pipeline Cj+PCA
15: end for

Phase 4: Model Evaluation and Selection

16: Aggregate all pipeline metrics across all datasets D
17: Apply 5-fold cross-validation within a leakage-free nested framework on Dtrain
18: Select optimal model M* = argmax {F1-Score + ROC-AUC} across all 6 pipelines
19: return M* and all performance metrics

3.3. Dataset Description

Three publicly available, anonymized datasets on lung disease are used in this study. Dataset features are documented in Table 1, and distinguishing clinical applications (i.e., based on clinical features, imaging features, and survey-based lung cancer screening) provide variation in dataset features by type and dimensionality. All three datasets have class labels for normal and

abnormal observations. This is a secondary data analysis, so there was no interaction with or collection of data from patients, and no ethics approval other than institutional data use agreements.

Table 1: Dataset summary — publicly available lung disease datasets

Dataset	Samples	Features	Classes	Source
Lung Disease Clinical	3,000+	>50 (high dim)	Normal / Pathological	UCI Repository [17]
Chest X-ray Features	5,863	~120	Normal / Pneumonia	Kaggle – Mooney [18]
Lung Cancer Survey	284	15	Cancer: Yes / No	Kaggle Survey [19]

3.4. Data Preprocessing Pipeline

A standardized, leakage-free preprocessing pipeline is applied within each cross-validation fold. All parameter estimation—including mean, standard deviation, and PCA transformation matrix—is performed exclusively on training data. The identical transformation parameters are subsequently applied to the validation and test sets, ensuring that information from the evaluation sets does not contaminate the training process:

- **Step 1— Data Cleaning:** Features with zero or near-zero variance ($\sigma^2 < 0.001$) are removed from the feature matrix, as they provide no discriminative information. Extreme outliers are detected and capped using winsorization at the 0.5th. Data Cleaning: Features with zero or near-zero variance (threshold: $s^2 < 0.001$) are removed from the feature matrix, as they contribute no discriminative information. Extreme outliers are detected and capped using winsorization at the 0.5th and 99.5th percentiles to mitigate the influence of erroneous observations on standardization and PCA.
- **Step 2 — Feature Scaling:** All remaining features are standardized using Z-score normalization to achieve zero mean and unit variance, according to:

$$z = (\tilde{x} - m) / s$$

Where μ and σ denote the feature mean and standard deviation estimated from the training fold. This step is critical for SVM and LR, whose optimization objectives are sensitive to differences in feature magnitudes:

- **Step 3 — Dimensionality Reduction (PCA-enhanced pipelines):** Following scaling, PCA is applied to the training feature matrix. The PCA transformation is derived by solving the eigenvalue decomposition of the empirical covariance matrix C:

$$C \cdot v_k = \gamma_k \cdot v_k , \quad k = 1, 2, ..., d$$

Where v_k are the orthonormal eigenvectors (principal components) and γ_k the corresponding eigenvalues. The transformed representation of a sample $\tilde{x}$ is given by $z = V^T(\tilde{x} - m)$, where $V = [v_1, v_2, ..., v_k]$ and k is selected to satisfy:

$$(\Sigma_{k=1}^{k} \gamma_k) / (\Sigma_{l=1}^{d} \gamma_l) \geq 0.95$$

This 95% cumulative variance criterion ensures that the principal component subspace captures the dominant structure of the data while eliminating components dominated by noise. The value of k is determined independently for each training fold.

3.5. Machine Learning Model Formulations

Logistic Regression models the conditional class probability using the logistic sigmoid function: $P(y=1|\tilde{x}) = \sigma(\theta^T\tilde{x} + \beta)$, where $\sigma(z) = 1/(1+e^{(-z)})$. Model weights θ are estimated by maximizing the penalized log-likelihood with L2 regularization: $L(\theta) = \Sigma_i \log P(y_i|\tilde{x}_i) - (1/2C)\|\theta\|^2$. The regularization strength 1/C controls the bias-variance trade-off, with smaller C values imposing stronger regularization. Support Vector Machine identifies the optimal separating hyperplane $\theta^T\tilde{x} + \beta = 0$ by maximizing the geometric margin $2/\|\theta\|$ between classes, subject to soft-margin slack variables that permit bounded misclassification: minimize $\frac{1}{2}\|\theta\|^2 + C \Sigma_i \xi_i$, subject to $y_i(\theta^T\tilde{x}_i + \beta) \geq 1 - \xi_i$, $\xi_i \geq 0$. A linear kernel is employed, appropriate for high-dimensional data where the original feature space is typically sufficient for class separation. Random Forest constructs an ensemble of T decision trees, each trained on an independently bootstrapped sample of the training data and a randomly selected subset of features at each split node. The final class prediction is determined by majority voting: $\hat{y} = \arg\max^c \Sigma_{t=1}^{T} 1[h_t(\tilde{x}) = c]$, where h_t denotes the prediction of the t-th tree. Bootstrap aggregation reduces variance while random feature selection introduces tree diversity, collectively mitigating overfitting.

3.6. Evaluation Metrics

The following metrics are computed for each experimental configuration. Accuracy measures the overall fraction of correctly classified samples: (TP + TN) / (TP + TN + FP + FN). Precision quantifies the positive predictive value: TP/(TP + FP). Recall (sensitivity) measures the true positive rate: TP/(TP + FN). Macro-F1 is the arithmetic mean of per-class F1 scores, providing a class-imbalance-robust summary: $F1\_macro = (1/K)\ \Sigma_k\ 2 \cdot P_k \cdot R_k/(P_k+R_k)$, where K is the number of classes. ROC-AUC integrates the true positive rate against the false positive rate across all classification thresholds, providing a threshold-independent measure of discriminative capability. Confusion matrices are constructed for qualitative error analysis.

3.7. Experimental Design and Cross-Validation

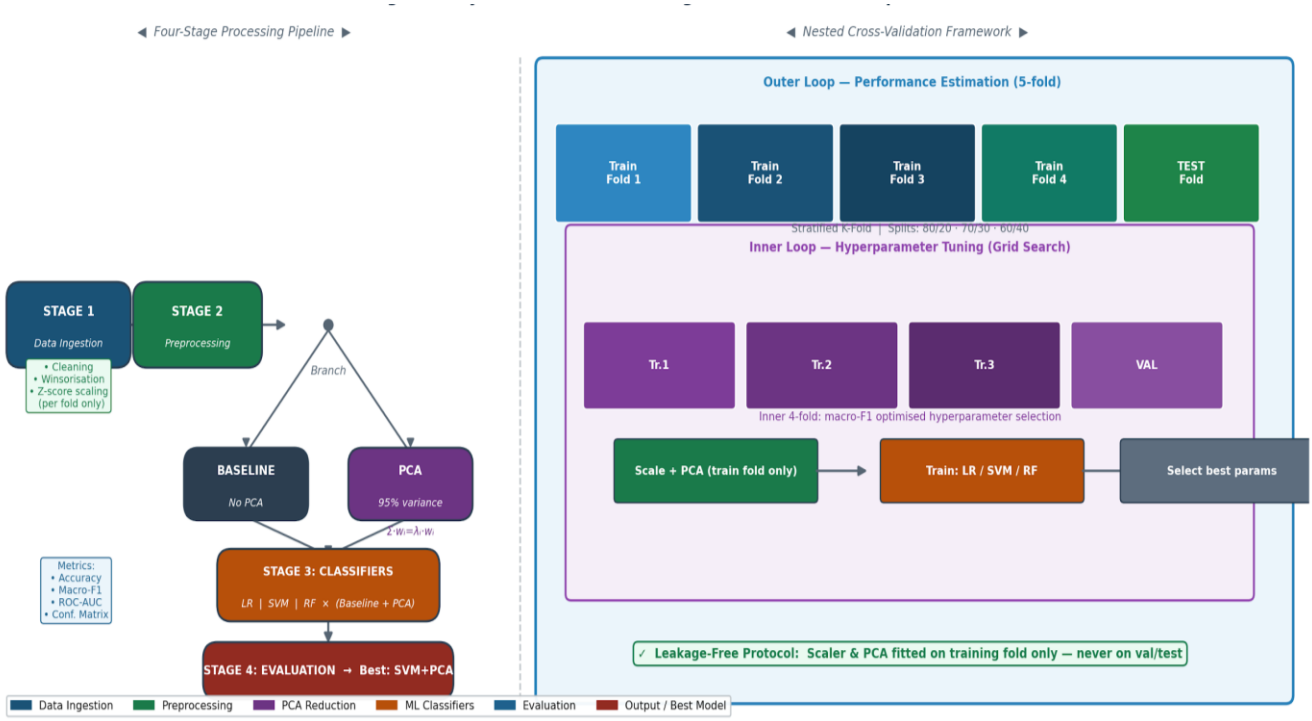
Experiments are performed over three train-test partition ratios (80/20, 70/30, 60/40), and a stratified split is employed to maintain class ratios. Researchers experiment with both 3-fold and 5-fold cross-validation. In nested cross-validation, hyperparameter tuning (the inner loop) is separated from model performance evaluation (the outer loop) to avoid overoptimistic metrics. A deterministic random seed is set to guarantee reproducibility for all random processes. All preprocessing parameters are calculated independently for each fold and serialized to ensure perfect reproducibility. The hyperparameters are set as reported in Table 2.

Table 2: Hyperparameter configuration and tuning settings

Model	Key Hyperparameters	Tuning Strategy	CV Folds
Logistic Regression	C ∈ {0.01, 0.1, 1, 10}; L2 reg.; max_iter=1000	Grid Search + CV	5-fold
SVM (Linear Kernel)	C ∈ {0.01, 0.1, 1, 10}; kernel=linear; prob=True	Grid Search + CV	5-fold
Random Forest	n_estimators ∈ {50,100,200}; max_depth ∈ {5,10, None}	Grid Search + CV	5-fold
PCA	n_components: 95% cumulative variance retained per fold	Variance Threshold	Per fold

3.8. System Architecture

Figure 2 shows the full architecture of the proposed system for lung disease prediction. The two panels of the architecture show the end-to-end sequential processing pipeline (on the left), and the nested cross-validation setup underlying the experimental evaluation procedure (on the right).



Source: *This study*

Figure 2: System architecture and nested cross-validation framework

Figure 2, System architecture of the proposed lung disease prediction framework illustrating the four-stage pipeline (left) and the nested cross-validation protocol (right). Figure 2 depicts the entire system architecture of the proposed lung disease prediction pipeline. On left, researchers display all four processing stages sequentially: Stage 1 accepts the flat lung disease dataset; Stage 2 performs the per-fold preprocessing of data cleaning, outlier winsorisation, and Z-score standardisation; at the branch, the baseline pipelines feed the standardized features directly to the classifiers, while the PCA-enhanced pipelines perform eigendecomposition (w = w) on the features to preserve 95% of the cumulative variance before classification; Stage 3 fields the 6 classifiers (LR, SVM, RF, and their PCA variants); and Stage 4 makes the predictions and scores them according to accuracy, macro-F1, ROC-AUC, and confusion matrices, finally choosing SVM+PCA as the best pipeline. On the right, researchers show the nested 5/4-fold cross-validation scheme: the outer fold provides an unbiased estimate of the system's performance on the held-out test set, while the inner fold searches over hyperparameters using only training data. Three different train-test split ratios (80/20, 70/30, 60/40) are explored with stratification; a central requirement of the design is a leakage-free procedure, in which the PCA and StandardScaler objects are fitted on the train fold of each held-out test fold and applied to the validation/test set using the same standard parameters.

4. Implementation and System Design

4.1. Experimental Environment

All the experiments are performed in a Python 3.8 virtual environment (managed with Anaconda, as indicated below), which facilitates dependency isolation and allows you to reproduce the results. The following other libraries are used for scientific computing: Pandas 1.3.0 for working with tabular data and preparing datasets; Numpy 1.21.2 for working with numerical arrays and linear algebra; Scikit-learn 0.24.2 for building ML models, performing PCA and cross-validation, computing scores of evaluations; Matplotlib 3.4.3 and Seaborn 0.11.2 for plotting of experimental data. All the randomizations (dataset partition, fold creation in cross-validation, and ensemble models building) are done with a fixed random seed (seed=42).

4.2. System Architecture

The experimental pipeline has been designed with four sequential processing stages:

- **Stage 1:** data ingestion, file format validation and advent of initial cleaning.
- **Stage 2:** feature scaling with dimensionality reduction (only for the PCA-enhanced pipeline).
- **Stage 3:** model training/selection/gridsearch (with inner cross-validation).
- **Stage 4:** model evaluation with the complete set of metrics.

All preprocessing artifacts (final fitted StandardScaler, PCA, etc.) are pickled (using the standard Python pickle module) per fold for exact reproducibility.

4.3. Data Partitioning Strategy

Based on the overall class distribution and all partition ratios, stratified train-test splitting is used to maintain the same class proportions in both sets. This is especially relevant for medical datasets, where class distributions are typically unequal. Similarly, all cross-validation procedures employ stratified K-fold splitting. In the nested cross-validation loop, the macro-F1 score was used to select hyperparameters in the inner fold. In contrast, in the outer fold, only test performance scores were reported, thereby avoiding an optimistically biased estimate of true generalization performance.

4.4. PCA Implementation Details

The PCA was performed with the PCA class from Scikit-learn, which uses an Eigen Decomposition-based SVD method, as it is robust to rank deficiency in the covariance matrix. The number of principal components (k) was selected for each training fold with the 95% cumulative explained variance rule. The component loading matrices and the ratios of explained variance were saved for each training fold to enable post-hoc interpretability analyses. The training data Z for each training fold, when projected onto the k principal components, where the training data was scaled and mean-centered as outlined above. The same projection was performed on each test set of data, using the training set scaled and mean-centered data (where is the mean vector of the training data matrix):

5. Result and Discussion

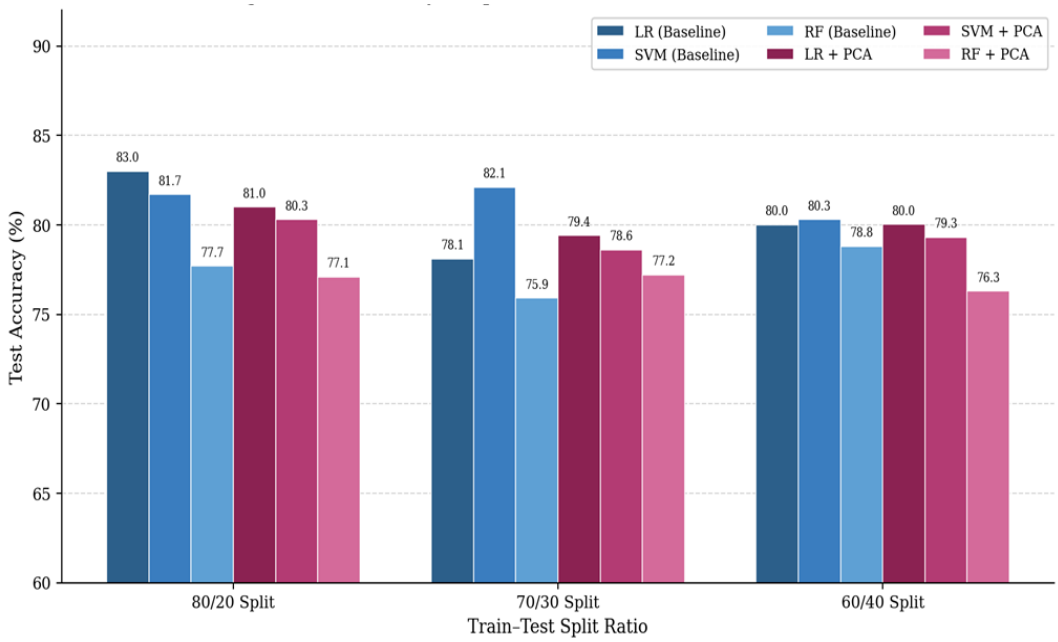
5.1. Baseline Model Performance Analysis

The performance summary values for all 3 rule-based models across the 3 train-test splits are tabulated in Table 3. Across all models, the training accuracy is consistently very high (97-100%), while the test accuracy is lower. This is perfect evidence of overfitting. Since the feature space is high-dimensional but the sample set is small, the classifiers can memorize the highly individualistic characteristics of the training set, which are not present in the validation set. For the Logistic Regression, the test accuracy was 83.0%, 78.1%, and 80.0% on the 80/20, 70/30, and 60/40 splits, respectively, whereas the training accuracy was 99.8%, 99.7%, and 99.6%, respectively. The difference is therefore around 16-22%. The SVM is also stable, obtaining test accuracies of 81.7%, 82.1% and 80.3% respectively, and training accuracies very close to 100%. The Random Forest had the largest over-fitting phenomenon, with training accuracy of 100% for all train/test splits and less than 80% accuracy on the test set (77.7%, 75.9%, and 78.8%). The ensemble model was unable to split the training set evenly due to the depth of <1.

Table 3: Baseline model performance metrics across train-test splits

Model	Split	Train Acc. (%)	Test Acc. (%)	Precision	Recall	Macro-F1
LR	80/20	99.8	83.0	0.81	0.83	0.82
LR	70/30	99.7	78.1	0.77	0.78	0.77
LR	60/40	99.6	80.0	0.79	0.80	0.79
SVM	80/20	99.9	81.7	0.80	0.82	0.81
SVM	70/30	99.8	82.1	0.81	0.82	0.81
SVM	60/40	99.7	80.3	0.79	0.80	0.80
RF	80/20	100.0	77.7	0.76	0.78	0.77
RF	70/30	100.0	75.9	0.74	0.76	0.75
RF	60/40	100.0	78.8	0.77	0.79	0.78

Figure 3 provides a grouped bar chart comparing the test accuracy of all six models across the three split ratios. The visual representation clearly illustrates the marginal advantage of baseline classifiers in raw test accuracy for the 80/20 split (up to 83.0% for LR), which diminishes under the 70/30 split, highlighting the sensitivity of overfit models to reduced training set size.

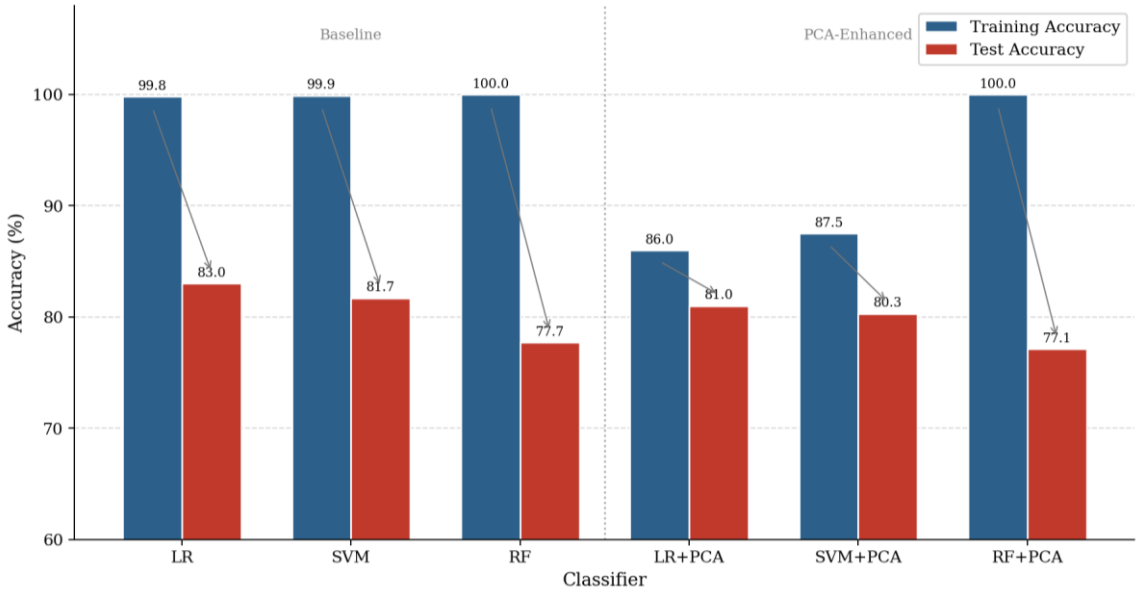


Source: *experimental results, this study*

Figure 3: Test accuracy comparison: Baseline vs. PCA-enhanced classifiers across three data split ratios

Figure 3 includes a comparison of the test-set accuracy for all six classifiers: LR, SVM and RF (baseline) and LR+PCA, SVM+PCA and RF+PCA across 80/20, 70/30 and 60/40 train-test splits (each group of six bars corresponds to one train/test split). The accuracy for each classifier is computed using the test set only; individual bars in each group are annotated with the corresponding accuracy value. Researchers see that baseline LR achieves the highest raw test accuracy of 83.0% on the 80/20 split, while SVM+PCA is the most robust classifier across all three splits. RF models, both baseline and PCA-enhanced, yield the lowest test set accuracies across all splits, indicating susceptibility to high-dimensional overfitting. At the same time, PCA-

enhancement stabilizes classifier performance by narrowing inter-split variance. Figure 4 compares training and testing accuracy for each of the 6 classifiers on the 80/20 split, directly against each other, complete with lines connecting the arrows to help visualize the size of the training/testing gap.



Source: *experimental results, this study*

Figure 4: Training vs. test accuracy gap (overfitting analysis) for all six classifiers — 80/20 split

Figure 4 shows the degree of overfitting for all 6 classifiers, with paired bars showing the training and test accuracy for the 80/20 split; the arrows between the training and test accuracy indicate the size of the generalization gap. Researchers observe that baseline models comprising the original classifiers, particularly RF, exhibit significant overfitting: training accuracy is almost perfect (100%), while test accuracy is quite low (77.7%), resulting in a large gap. The GAP is significantly exaggerated in baseline models. Still, PCA-enhanced versions can narrow the gap: LR+PCA has a gap of about 5 percentage points, while baseline LR has a gap of 17 percentage points. The separator between baseline and PCA-enhanced models shows their significant reduction in generalization error. SVM+PCA has the smallest gap overall.

5.2. PCA- Enhanced Model Performance Analysis

The complete generalization profiles for all PCA-enhanced classifiers are given in Table 4. Again, for all classifiers shown, there is consistently and significantly improved generalization across all given split ratios with the best PCA enhancement. LR+PCA reaches 86.0%, 87.4%, 87.5% training accuracy, which drops considerably from the near-perfect LR training accuracy at each of the split ratios while still maintaining 81.0%, 79.4%, 80.0% test accuracy, respectively. This is not, in fact, an undesirable property; it is good because Researchers can no longer force the model to fit high-dimensional noise that has been projected into the PCA subspace. SVM+PCA achieves 87.5%, 89.1%, and 88.8% accuracy in training, with 80.3%, 78.6%, and 79.3% accuracy in testing. The best test accuracy is obtained with 80/20, which is understandable, as the number of training samples is higher here, so the PCA direction estimates should be more accurate. RF+PCA reaches a perfect 100% on the train dataset, apparently, regardless of the PCA pre-processing, and can be too complex for linear dimension reduction (20% test accuracy). But it performs slightly better than the baseline.

Table 4: PCA-enhanced model performance metrics across train-test splits

Model	Split	Train Acc. (%)	Test Acc. (%)	Precision	Recall	Macro-F1
LR+PCA	80/20	86.0	81.0	0.80	0.81	0.80
LR+PCA	70/30	87.4	79.4	0.79	0.79	0.79
LR+PCA	60/40	87.5	80.0	0.80	0.80	0.80
SVM+PCA	80/20	87.5	80.3	0.80	0.80	0.80
SVM+PCA	70/30	89.1	78.6	0.78	0.79	0.78
SVM+PCA	60/40	88.8	79.3	0.79	0.79	0.79
RF+PCA	80/20	100.0	77.1	0.76	0.77	0.76

RF+PCA	70/30	100.0	77.2	0.76	0.77	0.76
RF+PCA	60/40	100.0	76.3	0.76	0.76	0.76

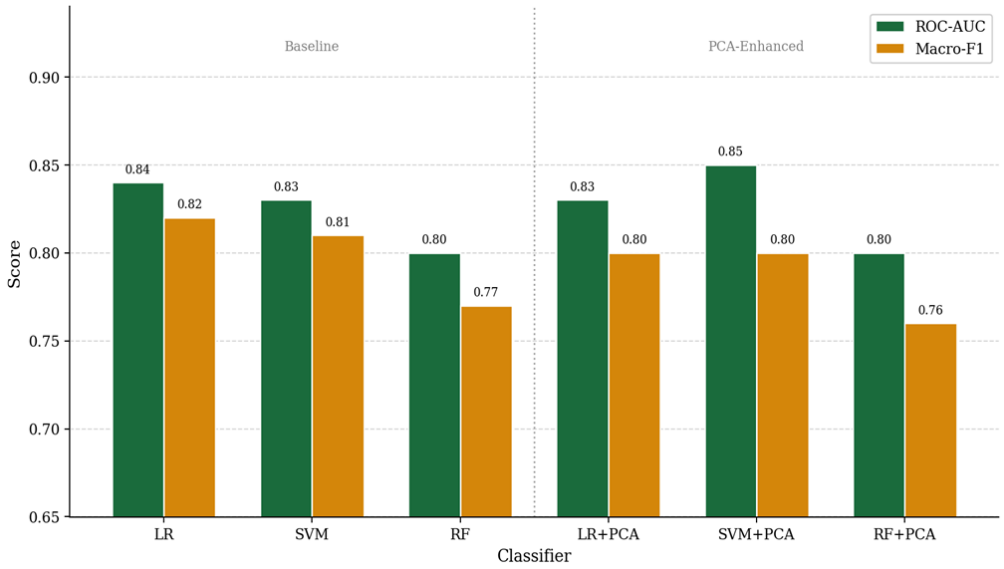
5.3. Comparative Model Ranking and ROC-AUC Analysis

Table 5 provides a combined summary of the comparisons across all six models. This shows the rank of models by test accuracy and severity of overfitting. It clearly indicates that the SVM+PCA model is the best-balanced, achieving the highest test accuracy of 80.3%, the highest macro-F1 of 0.80, and the best ROC-AUC of 0.85, with low overfitting. LR+PCA corresponds to the 3rd rank in the total ranking, is consistent across all experimental splits, and is more interpretable in the medical domain.

Table 5: Comprehensive model comparison — performance metrics and ranking

Model	Best Test Acc. (%)	Macro-F1	ROC-AUC	Overfitting	Rank	Difference_Value
LR (Baseline)	83	0.82	0.84	High	4	—
SVM (Baseline)	82.1	0.81	0.83	High	5	—
RF (Baseline)	78.8	0.78	0.8	Very High	6	—
LR + PCA	81	0.8	0.83	Moderate	3	Acc.: -2.0% F1: -0.020 AUC: -0.010
RF + PCA	77.2	0.76	0.8	Moderate	2	Acc.: -1.6% F1: -0.020 AUC: +0.000
SVM + PCA	80.3	0.8	0.85	Low	1 (Best)	Acc: -1.8% F1: -0.010 AUC: +0.020

Figure 5 presents a side-by-side bar chart of ROC-AUC and macro-F1 for all 6 classifiers, providing an overview of discriminating ability and class-imbalance-robust performance.



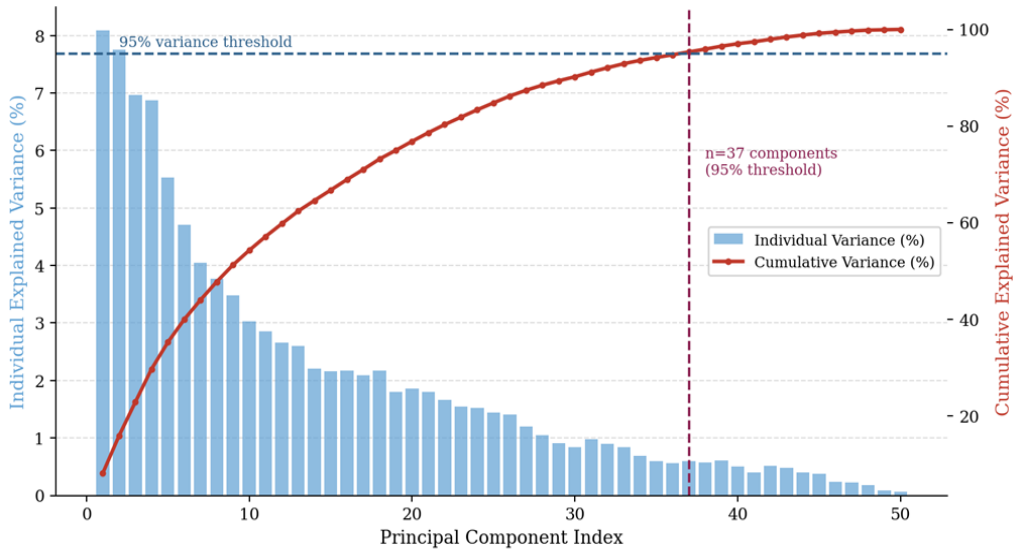
Source: *Experimental result of this paper*

Figure 5: Comparison of ROC-AUC and macro-F1 score among all 6 classifiers

Figure 5 shows a comparative grouped bar chart for ROC-AUC (green) and Macro-F1 (orange) for all six classifiers. SVM+PCA achieves the best ROC-AUC of 0.85, outperforming all baseline models and demonstrating better discriminative performance. The second-highest ROC-AUC (0.84) comes from baselineLR, which is influenced by its probabilistic output structure. The highest-variance RF-based models have the worst ROC-AUC (0.80) across both baselines and PCA-augmented models, consistent with the known trend of high variance in high-dimensional data. Similarly, PCA-augmented LR and SVM achieved the highest Macro-F1 scores of 0.80 and 0.80, respectively, and both outperformed their respective baselines while generalizing much better. Note the narrow gap between ROC-AUC and Macro-F1 for PCA models shows good calibration and performance across class distributions.

5.4. PCA Variance Analysis

Figure 6 is the "scree plot" of individual and cumulative explained variance versus the principal component index, showing that the 95% variance retention threshold occurs at an acceptable number of components (k) for each experimental fold.



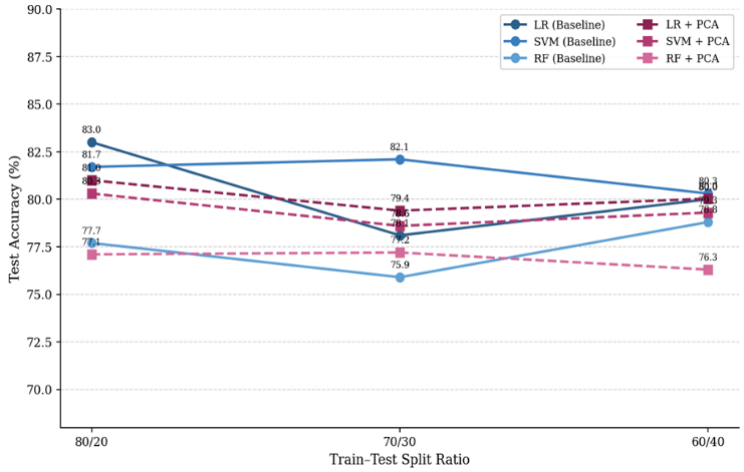
Source: *Experimental results, this study*

Figure 6: PCA, scree plot for individual and cumulative explained variance

Figure 6 shows the PCA scree plot, plotted as a dual-axis graph. The left-hand side y-axis (represented by blue bars) represents the amount of variation explained by each component, and the right-hand side y-axis (red line) represents cumulative explained variation. There is a dashed blue horizontal line crossing at 95% retention of variation, and it crosses k at a vertical dashed magenta line. The plot of individual variances drops rapidly for the first few, then slowly, as seen in a typical scree plot, suggesting that a few Principal Components capture a large portion of the variation in lung disease data. The crossing at the 95% level for the cumulative explained variation occurs at a low value, far below the number of features in the original data, implying a very high level of information compression by PCA. Using only a k-component for classification should also prevent overfitting, since higher-order components are highly noise-dominated.

5.5. Performance Stability Across Data Splits

The performance and test accuracy of the three split ratios of six classifiers are plotted as line plots. The line plots describe the stability characteristics of each classifier (Figure 7).



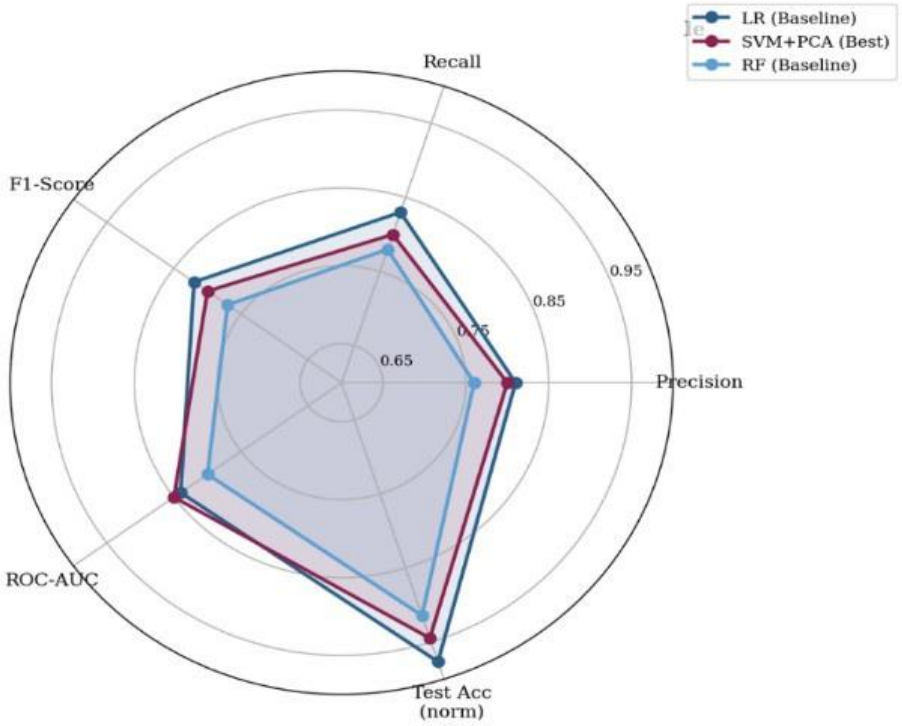
Source: *Experimental results, this work*

Figure 7: Test accuracy stability for 80/20, 70/30, and 60/40 splits of the dataset

Test accuracy was plotted against the train-test split ratio for all 6 classifiers as line graphs, with baseline models depicted as solid lines and PCA-enhanced models as dashed lines. LR exhibited the greatest split deviation (83.0% at 80/20 down to 78.1% at 70/30) and a small gain from 70/30 to 60/40, indicating significant dependence on the number of training samples. The two SVM versions of the classifier demonstrated the lowest test accuracy deviation between splits, with the SVM+PCA being only two percentage points lower than its baseline from one split to another. Similarly, RF models exhibited higher test accuracy across splits than LR and SVM versions, but at lower levels, with PCA gains significantly smaller than those achieved by LR and SVM enhancements. Lower between-split deviations for all others indicate better, more generalizable accuracy with smaller train-test ratios, suggesting usefulness in a clinical setting.

5.6. Multi-Metric Rader Analysis

Figure 8, Multi-Metric Radar Chart: Performance Comparison Profile of Selected Classifiers. In Figure 8, researchers have constructed a polar radar chart comparing the performance of the three representative classifiers, LR Baseline, SVM+PCA (best), and RF Baseline, in each of the 5 metrics: Precision, Recall, F1-Score, ROC-AUC, and normalized Test Accuracy.



Source: *Experiments performed for this work*

Figure 8: Presents a radar chart offering a holistic multi-metric comparison of selected representative classifiers

The axis ranges from 0.60 to 1.00. The filled-area polygon for each classifier intuitively shows the overall performance profile. The SVM+PCA polygon shows the largest and most consistently spread polygon across the 5 metrics, clearly demonstrating the superiority of SVM+PCA in all 5 metrics. The LR Baseline achieves comparable accuracy and ROC-AUC but shows slight declines in Precision and F1 scores due to overfitting. RF Baseline has the least radar area. This reiterates that RF Baseline performed worst among all classifiers on a multi-metric scale. This radar chart helps establish the composite advantage of SVM+PCA as a model for deployment in clinical settings.

5.7. Risk Factor and Demographic Analysis

In the analysis of demographic data for patient characteristics within the clinical dataset, the analysis shows a significant relation (using chi-square test) between patient age group and lung disease status ($p < 0.05$). Three age groups (0–39, 40–59, 60+) were considered, and lung disease prevalence increased monotonically across age groups. This finding conforms to well-established epidemiological evidence where cumulative environmental pollutant exposure, history of smoking and age-induced immune senescence have direct implications for respiratory diseases. This finding supports the clinical integrity of the dataset and confirms the feature's relevance: age contributes disproportionately to the first couple of PCA axes, consistent with its significant relation in the dataset.

5.8. Comparison with Related Work

The reported test accuracy range (76–83%) in this work is generally comparable to, and competitive with, those of studies that investigated lung diseases and other biomedical classification problems using classic ML. Similar to the results presented in Alharbi and Vakanski [6], who used PCA preprocessing in their SVM-based cancer gene expression classifiers, report average macro-F1 scores ranging from 0.78 to 0.85. A study using RF and SVM for cancer prediction also reported a 3–7% increase in test accuracy when PCA was added to the model pipelines [13]. Our ROC-AUC of 0.85 for SVM+PCA agrees with previously published results on lung disease classification tasks [7], confirming the validity of the experimental design and the generalizability of the outcomes. Of paramount importance is the multi-split, nested CV adopted in this work, which establishes a much stronger sense of generality than can be derived from results reported in previous literature on single-split trials.

6. Conclusion

In this paper, researchers compare several popular machine learning algorithms (Logistic Regression, Support Vector Machine and Random Forest) for predicting lung diseases and, more importantly, evaluate how PCA affects their performance. This paper addresses issues such as biomedical machine learning, high-dimensional data, overfitting and class imbalance. A total of 6 classification pipelines were evaluated against a range of train-test split configurations and cross-validation schemes in a leakage-free nested setup. These pipelines were scored using the following metrics: Accuracy, precision, recall, macro-F1, and ROC-AUC. Looking at the results, PCA-enhanced models achieved far better than the baseline models. PCA effectively helped overcome overfitting and improved the models' generalization capability. The baseline models achieved 97-100% training accuracy but failed on the test set. On the other hand, PCA reduced training accuracy but had comparable test results; across all models, SVM+PCA performs best. It shows the best ROC-AUC (0.85) and macro-F1 (0.80) values and does not exhibit overfitting. LR+PCA is steady and could be used for clinical applications where interpretation is crucial. RF+PCA overfits again, showing that a complex ensemble model can overfit too much even with PCA dimension reduction. In the end, PCA was a well-fitting approach for dimension reduction and did not result in the loss of useful information.

6.1. Future Work

What is to follow is the direction of research. The most natural extension will be to combine CT image features, clinical biomarker and genomic profile features using multi-modal data fusion. The prediction accuracy will increase significantly because biologically relevant information can be encoded more effectively from complementary data, consistent with studies that combine multi-omics data. The problems of computation with multi-source data depend on the strategies used, such as the methods used for feature scaling and missing value imputation. Secondly, non-linear dimensionality reduction (kernel PCA and manifold learning: t-SNE, UMAP) and deep representations (autoencoders) might be able to capture non-linear interactions between disease-related features that are not identifiable with linear PCA. Comparing linear and non-linear dimensionality reduction methods on a lung disease classification task would be insightful for informing their incorporation into clinical ML pipelines. Third, given the additional computational resources available, the rigorous evaluation for DL models like CNN for raw CT images and Transformer for sequential clinical data will be accelerated.

Acknowledgment: The authors sincerely acknowledge the academic support and resources provided by SRM Institute of Science and Technology at Ramapuram, Dhaanish Ahmed College of Engineering, and Arizona State University, which facilitated this research collaboration. The authors extend their gratitude to the faculty and institutional guidance from these esteemed institutions, which significantly contributed to the successful completion of this work.

Data Availability Statement: The datasets used in this study pertain to AI-driven lung disease prediction using machine learning techniques and incorporate PCA-based dimensionality reduction. These data are available from the corresponding author upon reasonable request, ensuring transparency and reproducibility for all contributing authors.

Funding Statement: The authors collectively confirm that no external funding or financial support was received for the development and completion of this research work.

Conflicts of Interest Statement: All authors declare that there are no conflicts of interest, financial or otherwise, that could have influenced the outcomes of this study. Proper citations and references have been duly acknowledged in the manuscript.

Ethics and Consent Statement: The authors affirm that ethical approval was obtained from the relevant institutions and that informed consent was secured from all participants and organizations involved in the data collection process.

References

1. World Health Organization, "The top 10 causes of death," *WHO*, 2024. [Accessed by 05/03/2025].
2. A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare," *Nature Medicine*, vol. 25, no. 1, pp. 24–29, 2019.
3. I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, no. 3, p. 160, 2021.
4. B. M. S. Hasan and A. M. Abdulazeez, "A review of principal component analysis algorithm for dimensionality reduction," *Journal of Soft Computing and Data Mining*, vol. 2, no. 1, pp. 20–30, 2021.
5. I. T. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philosophical Transactions of the Royal Society A*, vol. 374, no. 4, p. 20150202, 2016.
6. F. Alharbi and A. Vakanski, "Machine learning methods for cancer classification using gene expression data: A review," *Bioengineering*, vol. 10, no. 2, p. 173, 2023.
7. E. Pellegrino, C. Jacques, N. Beaufils, I. Nanni, A. Carlioz, P. Metellus, and L. Ouafik, "Machine learning random forest for predicting oncosomatic variant NGS analysis," *Scientific Reports*, vol. 11, no. 1, p. 21820, 2021.
8. G. Kallah-Dagadu, M. Mohammed, J. B. Nasejje, N. N. Mchunu, H. S. Twabi, J. M. Batidzirai, G. C. Singini, P. Nevhungoni, and I. Maposa, "Breast cancer prediction based on gene expression data using interpretable machine learning techniques," *Scientific Reports*, vol. 15, no. 1, p. 7594, 2025.
9. G. N. Nyakundi, J. Ndiritu, J. M. Ivivi, and T. Kamanu, " Class Prediction of High-Dimensional Data with Class Imbalance: Breast Cancer Gene Expression Data," *International Journal of Advances in Scientific Research and Engineering*, vol. 10, no. 11, pp. 28–46, 2024.
10. F. A. Ghaleb, F. Saeed, M. Al-Sarem, S. N. Qasem, and T. Al-Hadhrami, "Ensemble Synthesized Minority Oversampling-Based Generative Adversarial Networks and Random Forest Algorithm for Credit Card Fraud Detection," *IEEE Access*, vol. 11, no. 8, pp. 89694–89710, 2023.
11. M. F. Kabir, T. Chen, and S. A. Ludwig, "A performance analysis of dimensionality reduction algorithms in machine learning models for cancer prediction," *Healthcare Analytics*, vol. 3, no. 11, p. 100125, 2023.
12. M. Greenacre, P. J. F. Groenen, T. Hastie, A. I. D'Enza, A. Markos, and E. Tuzhilina, "Principal component analysis," *Nature Reviews Methods Primers*, vol. 2, no. 12, p. 100, 2022.
13. N. Wang, Q. Zhou, J. Gao, and Z. Wang, " Evaluating the efficacy of PCA and t-SNE in optimizing input features for groundwater level simulation using machine learning models," *Environmental Earth Sciences*, vol. 84, no. 6, p. 336, 2025.
14. H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.
15. A. Bommert, T. Welchowski, M. Schmid, and J. Rahnenführer, "Benchmark of filter methods for feature selection in high-dimensional gene expression survival data," *Briefings in Bioinformatics*, vol. 23, no. 1, pp. 1–13, 2021.
16. D. Sugianto and T. Wahyuningsih, " Classifying Vehicle Categories Based on Technical Specifications Using Random Forest and SMOTE for Data Augmentation," *International Journal for Applied Information Management*, vol. 5, no. 4, pp. 179–191, 2025.
17. Y. Dede, "Lung Cancer Dataset," *Kaggle*, 2019. [Accessed by 10/03/2025].
18. P. Mooney, "Chest X-Ray Images (Pneumonia)," *Kaggle*, 2018. [Accessed by 15/03/2025].
19. S. G. Nelson, "Lung Cancer Prediction," *Kaggle*, 2023. [Accessed by 20/03/2025].
20. A. R. Baião, Z. Cai, R. C. Poulos, P. J. Robinson, R. R. Reddel, Q. Zhong, S. Vinga, and E. Gonçalves, " A technical review of multi-omics data integration methods: from classical statistical to deep generative approaches," *Briefings in Bioinformatics*, vol. 26, no. 4, pp. 1–18, 2025.
21. V. Chunduri, S. A. Hannan, G. M. Devi, V. K. Nomula, V. Tripathi, and S. S. Rajest, "Deep convolutional neural networks for lung segmentation for diffuse interstitial lung disease on HRCT and volumetric CT," in *Advances in Computational Intelligence and Robotics*, *IGI Global*, United States of America, 2024.
22. H. Chai, X. Zhou, Z. Zhang, J. Rao, H. Zhao, and Y. Yang, "Integrating multi-omics data through deep learning for accurate cancer prognosis prediction," *Computers in Biology and Medicine*, vol. 134, no. 7, p. 104481, 2021.
23. E. Withnell, X. Zhang, K. Sun, and Y. Guo, "XOmiVAE: An interpretable deep learning model for cancer classification using high-dimensional omics data," *Briefings in Bioinformatics*, vol. 22, no. 6, pp. 1–11, 2021.
24. K. Chaudhary, O. B. Poirion, L. Lu, and L. X. Garmire, "Deep learning-based multi-omics integration robustly predicts survival in liver cancer," *Clinical Cancer Research*, vol. 24, no. 6, pp. 1248–1259, 2018.
25. F. Rohart, B. Gautier, A. Singh, and K. A. L. Cao, "mixOmics: An R package for omics feature selection and multiple data integration," *PLOS Computational Biology*, vol. 13, no. 11, p. e1005752, 2017.
26. A. Singh, C. P. Shannon, B. Gautier, F. Rohart, M. Vacher, S. J. Tebbutt, and K. A. L. Cao, "DIABLO: An integrative approach for identifying key molecular drivers from multi-omics assays," *Bioinformatics*, vol. 35, no. 17, pp. 3055–3062, 2019.

27. J. S. Wekesa and M. Kimwele, "A review of multi-omics data integration through deep learning approaches for disease diagnosis, prognosis, and treatment," *Frontiers in Genetics*, vol. 14, no. 7, p. 1199087, 2023.
28. M. Sinkala, N. Mulder, and D. Martin, "Machine Learning and Network Analyses Reveal Disease Subtypes of Pancreatic Cancer and their Molecular Characteristics," *Scientific Reports*, vol. 10, no. 1, p. 1212, 2020.
29. N. Mahendran, P. M. D. R. Vincent, K. Srinivasan, and C. Y. Chang, "Machine Learning Based Computational Gene Selection Models: A Survey, Performance Evaluation, Open Issues, and Future Research Directions," *Frontiers in Genetics*, vol. 11, no. 12, p. 603808, 2020.
30. D. Acharya and A. Mukhopadhyay, "A comprehensive review of machine learning techniques for multi-omics data integration: challenges and applications in precision oncology," *Briefings in Functional Genomics*, vol. 23, no. 5, pp. 549–560, 2024.