

Leveraging Machine Learning and Deep Learning Techniques for Improved Pathological Staging of Prostate Cancer

Raziehsadat Ghalamkarian^{1*†}, Marziehsadat Ghalamkarian^{1*†},
Morteza Ali Ahmadi², Sayed Mohammad Ahmadi³,
Abolfazl Diyanat⁴

^{1*}Biology, Islamic Azad University Najafabad, Isfahan, Iran.

²Computer Engineering, University of Qom, Qom, Iran.

³Computer Engineering, Iran University of Science and Technology,
Tehran, Iran.

*Corresponding author(s). E-mail(s): razieh.ghalamkarian2@gmail.com;
marzieh.ghalamkarian@gmail.com;

Contributing authors: ma.ahmadi@qom.ac.ir;
ahmadi.mohammad2008@gmail.com; adiyanat@iust.ac.ir;

[†]These authors contributed equally to this work.

Abstract

prostate cancer (Pca) continues to be a leading cause of cancer-related mortality in men, and the limitations in precision of traditional diagnostic methods—such as the Digital Rectal Exam (DRE), Prostate-Specific Antigen (PSA) testing, and biopsies—underscore the critical importance of accurate staging detection in enhancing treatment outcomes and improving patient prognosis. This study leverages machine learning and deep learning approaches, along with feature selection and extraction methods, to enhance PCa pathological staging predictions using RNA sequencing data from The Cancer Genome Atlas (TCGA). Gene expression profiles from 486 tumors were analyzed using advanced algorithms, including Random Forest (RF), Logistic Regression (LR), Extreme Gradient Boosting (XGB), and Support Vector Machine (SVM). The performance of the study is measured with respect to the F1-score, as well as precision and recall, all of which are calculated as weighted averages. The results reveal that the highest test F1-score, approximately 83%, was achieved by the Random Forest algorithm, followed by Logistic Regression at 80%, while both Extreme Gradient Boosting

(XGB) and Support Vector Machine (SVM) scored around 79%. Furthermore, deep learning models with data augmentation achieved an accuracy of 71.23%, while PCA-based dimensionality reduction reached an accuracy of 69.86%. This research highlights the potential of AI-driven approaches in clinical oncology, paving the way for more reliable diagnostic tools that can ultimately improve patient outcomes.

Keywords: prostate cancer, Machine learning, Deep learning, Pathological staging, feature selection, feature extraction

1 Introduction

1.1 Background

prostate cancer (Pca) is the foremost cause of cancer-related deaths among men, with around 1,466,680 new cases and 396,792 deaths reported globally each year [5]. PCa is a multifactorial disease shaped by numerous risk factors. Key nonmodifiable risk factors include age, race, family history, and germline mutations. In contrast, metabolic syndrome, obesity, and smoking are considered potential modifiable risk factors. Furthermore, a variety of environmental, lifestyle, infectious, and dietary factors may also play a role in the development of PCa, although the evidence backing these links is generally weak [11]. The accurate assessment of pathological staging in PCa facilitates tailored treatment approaches and enhances the probability of improved therapeutic outcomes, thereby increasing survival rates and mitigating the financial and psychological burdens faced by patients [26].

1.2 Research study motivation

Accurate staging of PCa is essential, as it influences the selection of appropriate patient treatment options. However, many existing methods for detecting and staging PCa do not provide adequate accuracy.

- ✓ *Digital Rectal Exam (DRE)*: Although it is frequently utilized, it is not a dependable diagnostic instrument. This is due to the lack of consistent observable characteristics that can clearly differentiate between benign and malignant nodules [16].
- ✓ *Prostate-Specific Antigen (PSA) Testing*: PSA, which is the most reliable marker for screening PCa; nonetheless, its effectiveness in staging is limited due to significant overlap between PSA levels and tumor stages [25].
- ✓ *Biopsy*: The reliability of biopsy is low due to the limited accuracy in determining the histological grade, as evidenced by only 41% agreement with the grades obtained from prostatectomy [27].
- ✓ *Transrectal Ultrasound (TRUS)*: The accuracy of TRUS in detecting PCa is notably low. While TRUS can identify hypoechoic lesions in the peripheral zone, the variability in appearances often leads to considerable overlap with benign lesions, resulting in low sensitivity and specificity for PCa diagnosis [14]. As a result of the shortcomings of current diagnostic methods, it is crucial to identify reliable parameters for the detection and staging of PCa.

In recent years, there has been increasing interest in omics technologies within life sciences and clinical analysis. These methods offer crucial insights into the pathogenesis of diseases at the metabolite, protein, and transcriptome levels.

Transcriptomics involves the comprehensive analysis of an organism’s transcriptome, which encompasses all RNA transcripts. This field has been utilized to gain insights into the nature of diseases and to identify diagnostic and prognostic biomarkers. Additionally, high-throughput RNA sequencing (RNA-seq) data offers the chance to examine numerous transcripts, providing a thorough understanding of the expression dynamics associated with diseases [8].

RNA-Seq data is readily available from various databases and is being used to categorize patients based on their cancer stage. [4]. However, analyzing RNA gene expression data is quite intricate due to its high dimensionality, complexity, and the presence of duplicate feature values [9]. Therefore, there is a need for automatic feature extraction, which can be addressed using Artificial intelligence (AI) [18]. AI includes various technologies that share the common goal of computationally mimicking human intelligence. Machine learning is a subset of AI that identifies relationships within data by uncovering underlying patterns based on previous experiences and learning [1].

Deep learning is a subset of machine learning that emphasizes making predictions through multi-layered neural network algorithms modeled after the brain’s neurological structure. In contrast to other ML techniques like logistic regression, the neural network framework of DL allows models to scale exponentially as the volume and complexity of data increase. This makes deep learning especially effective for addressing complex computational challenges, including large-scale image classification, natural language processing, and speech recognition and translation [20].

Recently, the applications of AI in medicine have broadened to encompass not just clinical research, but also translational medicine and clinical procedures for a range of diseases, including cancers [17].

1.3 Literature Review

Several studies have demonstrated the efficacy of AI algorithms in analyzing medical imaging and genomic data to improve diagnostic accuracy and prognostic assessments. For instance, research by [13] revealed a novel methodology for analyzing genomic activity across three laterality classes: left, right, and bilateral. Utilizing a dataset of 450 samples, the researchers successfully identified key genes that serve as indicators for each class, achieving an impressive accuracy rate of nearly 99%. Among the significant differentially expressed genes are RTN1, HLA-DMB, and MRI1, which effectively distinguish between the classes and correlate with disease progression[13]. Additionally, [28] investigated the variability in aggressiveness of prostate cancer by analyzing a dataset of 106 RNA-Seq samples. The study employed machine learning techniques to identify 44 differentially expressed transcripts associated with cancer progression. Notably, the transcripts USP13 and PTGFR were found to have reduced expression in advanced stages, correlating with findings in other cancers like breast and ovarian [28].

The authors in [19], introduced a novel histopathological dataset for prostate cancer detection, comprising over 2.6 million tissue patches from 430 fully annotated scans. The dataset included 4675 scans with binary diagnoses and 46 scans evaluated by histopathologists. They also presented a machine learning framework for identifying cancerous tissue regions and predicting scan-level diagnoses, employing thresholding to handle uncertain cases. Their approach, which utilized ensembles of deep neural networks at various scales, achieved 94.6% accuracy in patch-level recognition and demonstrated high statistical agreement with nine human histopathologists in scan-level diagnosis[19]. [29] examined the role of AI and machine learning in managing prostate cancer, highlighting the current trends and future possibilities. AI has improved digital pathology, facilitating quicker and more precise diagnoses, enhancing lesion detection, and predicting patient outcomes. It has also contributed to predicting

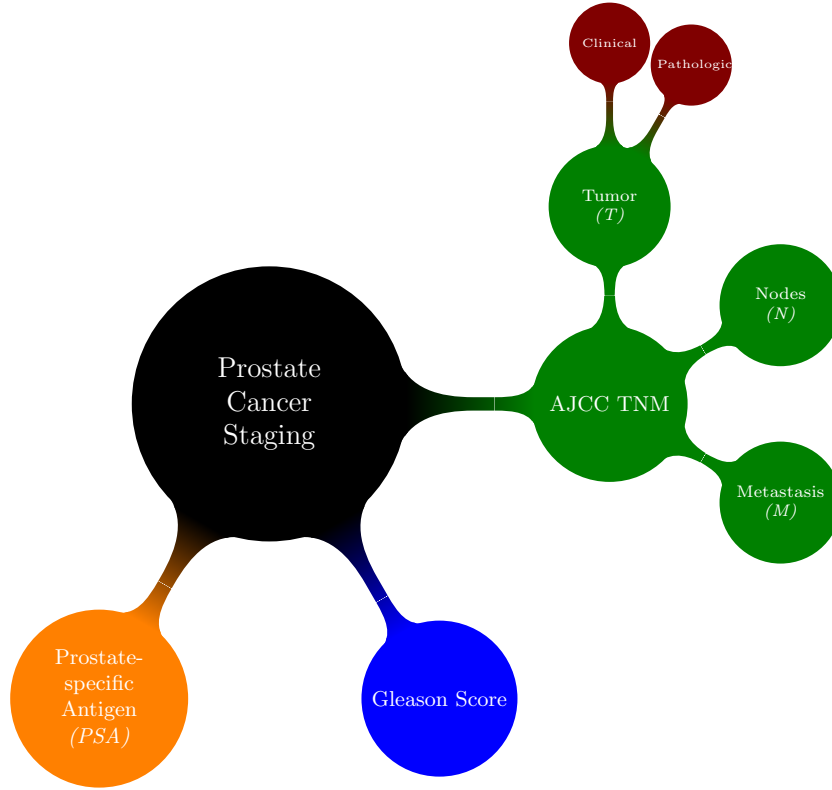


Fig. 1 Three methods of prostate cancer staging: the Prostate-specific Antigen (PSA), Gleason Score, and the AJCC TNM system, where AJCC stands for the American Joint Committee on Cancer. This system categorizes the cancer based on Tumor (T) size and extent, Nodes (N) for lymph node involvement, and Metastasis (M) for the presence of distant spread.

radiotherapy toxicity and increasing the autonomy of surgical robots for independent problem-solving [29].

AI has revolutionized the detection of prostate cancer with MRI images, [23] presented a fully automated deep learning system designed for the analysis of MRI-visible lesions. This system leveraged data collected from two distinct institutions. The lesions were meticulously annotated by a qualified radiologist, and subsequent biopsies were performed to establish a reliable ground truth, which was utilized for training the UNet and AH-Net architectures. The dataset comprised 525 patients, split into training ($n = 368$), validation ($n = 79$), and test ($n = 78$) cohorts. Results indicated that AHNet outperformed UNet in validation sensitivity (74.4% vs. 70.9%) and false positives (0.87 vs. 1.41). In the test cohort, UNet achieved a sensitivity of 72.8% compared to AHNet's 63.0%. Overall, the DL-based AI system shows promise for aiding radiologists in detecting prostate cancer lesions, despite challenges with false positives[23].

1.4 Contribution

This study advances prostate cancer staging by utilizing machine learning and deep learning techniques to effectively differentiate between early and late clinical stages. By analyzing gene expression data from RNA-sequencing alongside AJCC pathologic tumor stage information from The Cancer Genome Atlas (TCGA), we developed robust supervised classification models, with the Random Forest algorithm achieving an impressive F1-score of approximately 83%. This highlights the potential of advanced algorithms to enhance diagnostic accuracy compared to traditional methods.

Additionally, our research addresses high-dimensional data and class imbalance challenges through innovative strategies like stochastic feature augmentation and the Synthetic Minority Over-sampling Technique (SMOTE). These approaches improved model robustness and ensured accurate identification of cancer stages. By showcasing the efficacy of AI-driven techniques in clinical oncology, this study lays the groundwork for developing more dependable diagnostic tools and tailored treatment approaches, thereby improving patient outcomes in the management of prostate cancer.

1.4.1 Highlight

- ❶ **Need for Improvement:** Prostate cancer PCa is a leading cause of cancer-related deaths, emphasizing the need for enhanced diagnostic accuracy.
- ❷ **Objective:** Improve pathological staging of PCa using machine learning and deep learning techniques.
- ❸ **Data Source:** RNA-sequencing data and pathologic tumor stage information from The Cancer Genome Atlas (TCGA).
- ❹ **Methods Used:**
 - **Algorithms:** Random Forest, Decision Tree, Naïve Bayes, Logistic Regression, Support Vector Machine, K-Nearest Neighbors, XGBoost and Neural Networks.
 - **Dimensionality Reduction:** Principal Component Analysis and Independent Component Analysis were utilized to simplify high-dimensional gene expression data.
 - **Class Imbalance Mitigation:** The SMOTE was applied to address dataset imbalance, improving model robustness.
 - **Data Augmentation Strategies:** Stochastic Feature Augmentation (SFA) enhanced model generalization and predictive accuracy.
- ❺ **Key Findings:** The Random Forest algorithm achieved an F1-score of 83%, demonstrating improved diagnostic precision and reliability.
- ❻ **Clinical Implications:** Highlights the potential of AI-driven approaches to develop robust diagnostic tools and personalized treatment strategies in oncology.

2 Materials and Methods

2.1 Gene Expression Data Matrix

RNA-sequencing (RNA-seq) data related to gene expression for the AJCC (American Joint Committee on Cancer) pathological T stage, which indicates the size and

extent of the primary tumor, was obtained from the TCGA data portal (<http://cancergenome.nih.gov>) for 486 primary prostate cancer tumors.. The dataset includes 60,660 genes, offering insights into the overall RNA expression within the tumors. As a result, the data matrix consists of 60,660 columns and 486 rows. The gene expression values were normalized using Transcripts Per Million (TPM) to account for variations in sequencing depth among the samples. Additionally, the pathological T stages ($t1a, t1b, t1c, t2, t2a, t2b, t2c$) were labeled as *early stage*, while ($t3a, t3b, t4$) were labeled as *late stage*.

2.2 Differential expression analysis

A differential gene expression analysis was conducted to evaluate changes between the early (*stag1*) and late (*stag2*) stages of the condition. The gene expression data was loaded and relevant columns were extracted, specifically excluding case IDs and stage information. The dataset was divided into two distinct groups based on the stage. For each gene, the log2 fold change (log2FC) was calculated by comparing the mean expression levels in the late stage to those in the early stage. To assess the statistical significance of the observed differences, independent two-sample t-tests were performed for each gene, resulting in p-values compiled alongside the corresponding log2FC values. A significance threshold of 0.05 was established, with genes classified as significantly up-regulated if the log2FC was greater than 1.0 and as down-regulated if the log2FC was less than -1.0. For visual representation, a volcano plot was generated, illustrating log2FC on the x-axis and negative log10 p-values on the y-axis, with points color-coded to reflect their significance status.

2.3 Feature Selection

Feature selection is an essential initial step in numerous machine learning workflows. It significantly enhances the model's performance, interpretability, and efficiency by concentrating on the most important elements of the data while removing irrelevant or redundant information. In this study, we utilize SelectFpr for feature selection.

2.3.1 SelectFpr

SelectFpr is a feature selection method in machine learning that identifies features based on a predetermined false positive rate. It belongs to the Select family of feature selection techniques available in the scikit-learn library in Python. The SelectFpr algorithm operates by choosing features that exhibit a false positive rate below a defined threshold. It employs statistical tests, such as the chi-squared test or ANOVA F-test, to assess the relationship between each feature and the target variable, selecting those that satisfy the specified false positive rate criteria [2].

2.4 Feature Extraction

Real-world data with a large number of dimensions (attributes) demands greater time and computation at each step from data cleaning to model development. This complexity also complicates visualization efforts. Feature extraction techniques are employed

to address these challenges. The primary goal of feature extraction methods is to reduce the dataset's size while preserving as much of the original data content as possible. In essence, this involves identifying and eliminating non-essential components of the data to achieve dimension reduction. In this study, Independent Component Analysis (ICA) is utilized, which will be discussed in detail.

2.4.1 ICA

ICA is a statistical technique that aims to represent multivariate data as linear combinations of independent components. It is assumed that multivariate data is formed from a linear combination of a set of independent factors. Typically, the number of these factors is considered to be equal to the number of variables.

The dataset comprises p variables, each sampled at n points, represented by the Z matrix. In this scenario, the Z matrix in the ICA model is computed as indicated in below equation.

$$Z = AY. \quad (1)$$

A represents the mixing matrix, while Y denotes the source matrix that contains the independent components. Both the mixing matrix and the source matrix are considered unknown. Both matrices are estimated by maximizing the statistical independence of the predicted components, relying solely on the Z data matrix. Initially, the mixing matrix A is estimated. Subsequently, the Y matrix is derived using the estimated matrix A as follow.

$$Y = A^{-1}Z. \quad (2)$$

As a result, the independent components are identified, and the data is represented as a linear combination of these components [30]

2.4.2 PCA

Principal Component Analysis (PCA) is a technique primarily utilized in machine learning, functioning as a method for dimensionality reduction. It transforms a dataset with a larger number of features into one with fewer features, while maintaining the integrity of the data structure and keeping only the most significant attributes. PCA can be summarized in four steps within the context of machine learning. After obtaining the data, the first step involves normalizing it, which means adjusting each feature to have a mean of 0 and a standard deviation of 1. The second step entails constructing the Covariance Matrix, a square matrix that represents the correlations between features. The Covariance Matrix can be defined as:

$$M^{nn} = m_{ij}, \quad (m_{ij} = \text{Cov}(A_i, A_j)). \quad (3)$$

Also we have,

$$\text{Cov}(A_i, A_j) = \frac{\sum_{k=1}^n (A_{ik} - \bar{A}_i)(A_{jk} - \bar{A}_j)}{n - 1}. \quad (4)$$

$\text{Cov}(A_i, A_j)$ represents the covariance between the attributes A_i and A_j of the dataset based on n observations. \bar{A}_i and \bar{A}_j are the arithmetic means of A_i and A_j . with the

covariance Matrix. One way to define eigenvectors of a matrix M is a vector u , which satisfies the following equation.

$$\mu = \delta u. \quad (5)$$

δ represents a scalar that corresponds to the eigenvalue associated with the eigenvector. The final step involves ordering the eigenvectors from highest to lowest value and disregarding the less significant components, selecting only the most relevant ones [22].

2.5 Data Augmentation

2.5.1 Stochastic Feature Augmentation (SFA)

SFA is introduced to enhance feature representations by incorporating random noise. Specifically, we augment the latent feature embeddings by multiplying them with random variables and adding noise. The augmentation function is defined as:

$$\hat{z}_i = A(z_i) = \alpha \odot z_i + \beta. \quad (6)$$

Here, α and β are noise samples, and \odot represents element-wise multiplication. The values of α are drawn from Normal distributions $N(\mu, \Sigma)$, where the statistical parameters μ and Σ can be treated as hyperparameters or updated during model training based on feature statistics.

2.5.2 Simple Data-independent Noise

We first explore a basic version of SFA, where we set μ as a constant and Σ as a fixed diagonal matrix. This formulation allows for random perturbations of the original feature representations without favoring specific directions. The scale α and bias β are sampled from multivariate Gaussian distributions:

$$\alpha \sim \mathcal{N}(1, \sigma_1 I), \quad \beta \sim \mathcal{N}(0, \sigma_2 I) \quad (7)$$

In this context, I is the identity matrix, and σ_1 and σ_2 are scalar hyperparameters. By default, we set $\sigma_1 = \sigma_2$ to simplify the model. Although this approach does not target specific perturbation directions, it has been shown to significantly enhance the performance of the base Empirical Risk Minimization (ERM) method. To ensure the independence of the noise, we add a Gaussian noise with a very small standard deviation to the original data. The noise distribution is defined as:

$$N \sim \mathcal{N}(\mu, \sigma^2). \quad (8)$$

The formulation for creating the noise is:

$$N = \mu + \sigma Z, \quad (9)$$

where $Z \sim \mathcal{N}(0, 1)$ is a standard normal random variable, and σ is very small. Finally, the relationship between the original data X and the new data Y can be expressed as:

$$Y[i] = X[i] + N[i], \quad \forall i \in \{1, 2, \dots, n\}, \quad (10)$$

where X represents the original data and Y represents the new augmented data.

2.6 Oversampling Method for Imbalance Dataset

Machine learning techniques can encounter challenges when one class significantly outnumbers the others in a dataset, resulting in what is known as an imbalanced dataset. This imbalance can mislead the classification process, adversely affecting the ability to identify the minority class. In this study, we employed SMOTE to address the issue of imbalanced datasets.

2.6.1 Synthetic minority oversampling technique (SMOTE)

SMOTE is a method that randomly generates new instances of the minority class by sampling from the nearest neighbors within that class. These instances are created based on the features of the original dataset, with the goal of resembling the existing minority class instances. The equation below is utilized by SMOTE to synthetically boost the minority class (11).

$$X_{\text{new}} = X + \partial(X_i - X), \quad \partial \in (0, 1) \quad (11)$$

In this context, X_{new} denotes a new synthetic sample, while X refers to the feature vector of each sample in the minority class. The term X_i represents the i -th nearest neighbor of the sample X . Additionally, ∂ is a random value that falls between 0 and 1.

2.7 Classification Algorithms

Seven advanced supervised machine learning algorithms are employed alongside grid search cross-validation (CV), including Random Forest, Decision Tree, Naïve Bayes, Logistic Regression, Support Vector Machine, K-Nearest Neighbors, and XGBoost. Grid search cross validation (CV) was employed to optimize hyperparameters and fit the model to the training data using the best parameters. This study utilizes k -fold cross-validation, with the number of folds set to 5.

2.7.1 Random Forest (RF)

It is a meta-estimator that builds multiple decision tree classifiers from subsets of a dataset and uses averaging to improve accuracy and reduce the risk of overfitting [21].

2.7.2 Decision Tree (DT)

A decision tree is a predictive model that can be visualized as a tree, where each branch represents a classification question and the leaves indicate partitions of the dataset along with their classifications. Decision trees are effective classification algorithms that are gaining popularity with the rise of data mining in information systems. Well-known decision tree algorithms include Quinlan’s ID3, C4.5, C5, and Breiman et al.’s CART. Decision trees can be utilized for segmenting the original dataset, with each segment corresponding to a leaf of the tree. This allows for the grouping of similar data [15].

2.7.3 K-Nearest Neighbors (KNN)

It is another popular algorithm for classification tasks. This nonparametric algorithm does not make assumptions about the data. Its core concept is to classify data using a distance metric, like Euclidean or Manhattan distance. Choosing the best value of K for the nearest neighbors can be challenging, as there are no clear methods for finding the optimal K. A small K may result in unstable decision boundaries, whereas a larger K could introduce considerable bias [7].

2.7.4 Logistic Regression (LR)

It is a commonly used algorithm for binary classification that estimates the likelihood of a given instance belonging to a particular category. The logistic function, or sigmoid function, transforms predicted values into probabilities [10].

2.7.5 Naïve Bayes (NB)

It is a probabilistic machine learning algorithm based on Bayes' Theorem, which assumes that features are conditionally independent when the class label is known [32].

2.7.6 Support Vector Machine (SVM)

It is a classification algorithm that separates datasets into different classes by identifying a maximum marginal hyperplane (MMH) based on the closest data points [31].

2.7.7 Extreme Gradient Boosting (XGB)

At its core, XGBoost is a boosting algorithm based on decision trees. Boosting is an ensemble learning method that involves constructing multiple models in sequence, where each subsequent model aims to address the shortcomings of the prior one. In tree boosting, each new model added to the ensemble is a decision tree [24].

2.7.8 Neural network

A sequential neural network is a linear stack of layers, ideal for classification and regression tasks. It includes an input layer that receives data, one or more hidden layers for computations, and an output layer for predictions.

In this model, each neuron in one layer connects to every neuron in the next, with weights assigned to each connection. Neurons compute a weighted sum of their inputs followed by an activation function, like ReLU or Sigmoid, to introduce non-linearity. During training, the model adjusts its weights using backpropagation to minimize prediction errors [6].

2.7.9 Grid search cross-validation for hyperparameter tuning

With the appropriate combination of hyperparameters, it is possible to create a robust and accurate machine learning model. Hyperparameter tuning involves selecting the best set of parameters. To enhance performance metrics, the dataset should be trained

using various machine learning techniques and different hyperparameter combinations. The dataset can be processed through multiple machine learning methods using the cross-validation (CV) technique. Below are some key terms to consider when utilizing grid search CV (GridSearchCV):

Estimator: In scikit-learn, this term refers to the interface used to set up the estimator. This parameter specifies the classifier that needs to be trained.

Parameter Grid: This consists of parameter names and their corresponding settings organized in a Python key-value dictionary. All parameters are evaluated for optimal results.

CV: This defines the approach for splitting the data. Cross-validation is a resampling technique that is used to evaluate machine learning models. Its primary goal is to determine how well the models perform on unseen data. The process begins with randomly shuffling the dataset, followed by creating k groups from the entire data set. While one group serves as the test set, the remaining groups are used for training. Each sample is used k-1 times, appearing only once in the testing results. different Hyperparameter with Range of Values for each algorithm [Table 1](#).

Table 1 Hyperparameters for Different Methods

Method	Hyperparameter	Range of Values
RF	n_estimators	50/100/200
	max_depth	None/10/20/30
	min_samples_split	2/5/10
DT	criterion	gini/entropy
	max_depth	None/10/20/30
	min_samples_split	2/5/10
	max_features	4/6/8
	min_samples_leaf	1/2/4
KNN	n_neighbors	3/5/7/9/11
	weights	uniform/distance
	metric	euclidean/manhattan
LR	C	0.001/0.01/0.1/1/10/ 100
	penalty	l1/l2/elasticnet/None
	solver	liblinear/saga
	max_iter	100/200/300
NB	var_smoothing	1e-9/1e-8/1e-7/1e-6
SVM	C	0.1/1/10/100
	kernel	linear/poly/rbf/sigmoid
	gamma	scale/auto
XGBoos	n_estimators	50/100/150
	max_depth	3/5/7
	learning_rate	0.01/0.1/0.2

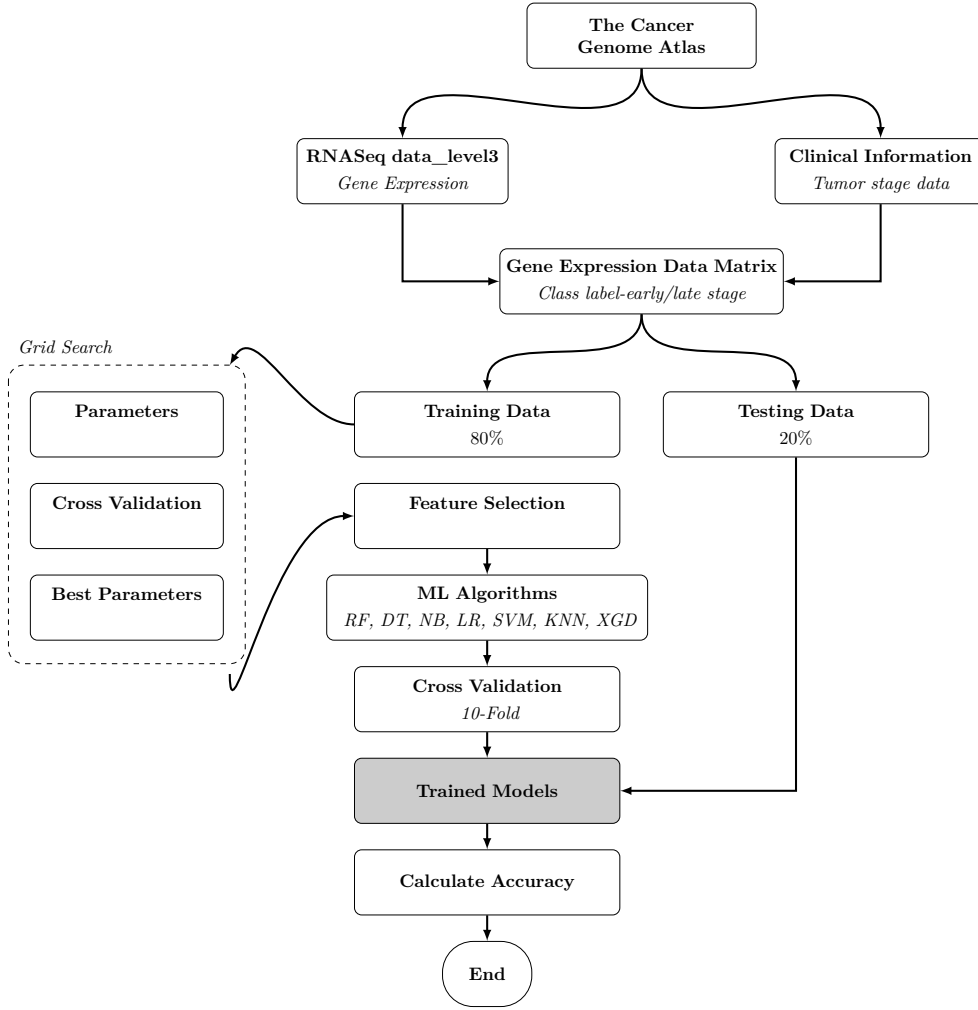


Fig. 2 Overview of the strategy employed in this paper. Data was obtained from the TCGA data portal, where gene expression values from tumors served as the descriptor variables. Class labels were derived from the clinical details provided in the Biotab section of the TCGA data portal. The processed data was then used to create training and testing datasets. Feature selection and extraction techniques were applied, and classification models were developed using the scikit-learn library in Python.

2.8 Evaluation measures

The confusion matrix is a commonly utilized tool for tackling classification problems, applicable to both binary and multiclass scenarios. This $N \times N$ -dimensional matrix displays a classifier's performance across various classes when evaluated against a set of

Table 2 Overview of the total number of instances in the training and testing dataset.

Class label	Total samples	Training	Testing
Early stage	184	146	38
Late stage	302	242	60
Total	406	324	82

test data, though it necessitates known actual values for the test data to be effective. Confusion matrices provide a summary of the predicted value counts compared to actual values. Several evaluation criteria are listed below:

Precision It describes the proportion of accurately predicted positive observations by the system relative to the total number of positive observations predicted (12).

$$\text{Precision} = \frac{TP}{TP + FP} \times 100 \quad (12)$$

Recall It is the ratio of accurately predicted positive observations by the system to the total number of actual positive instances, and it is also known as sensitivity (13).

$$\text{Recall} = \frac{TP}{TP + FN} \times 100 \quad (13)$$

F1-Score It is the weighted average of precision and recall, taking into account both false positives (FP) and false negatives (FN). This metric offers a measure of the balance between recall and precision (14).

$$\text{F1-score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \times 100 \quad (14)$$

3 Results and Discussions

3.1 Gene expression data matrix

The RNAseq dataset, which comprises a matrix with dimensions of $486 \times 60,660$ for rows and columns, was retrieved from the TCGA data portal for 486 pathologically diagnosed PCa patients. This data set includes gene expression levels for 60,660 genes in the primary tumor tissue samples from patients. Furthermore, the study dataset was divided into 80% for training and validation and 20% for independent testing. The distribution of patients by pathologic tumor stage in both the training and the testing datasets is presented in Table 2.

3.2 Differential expression analysis

The differential gene expression analysis revealed significant variations between the early (stag1) and late (stag2) stages of the condition, as illustrated in the volcano

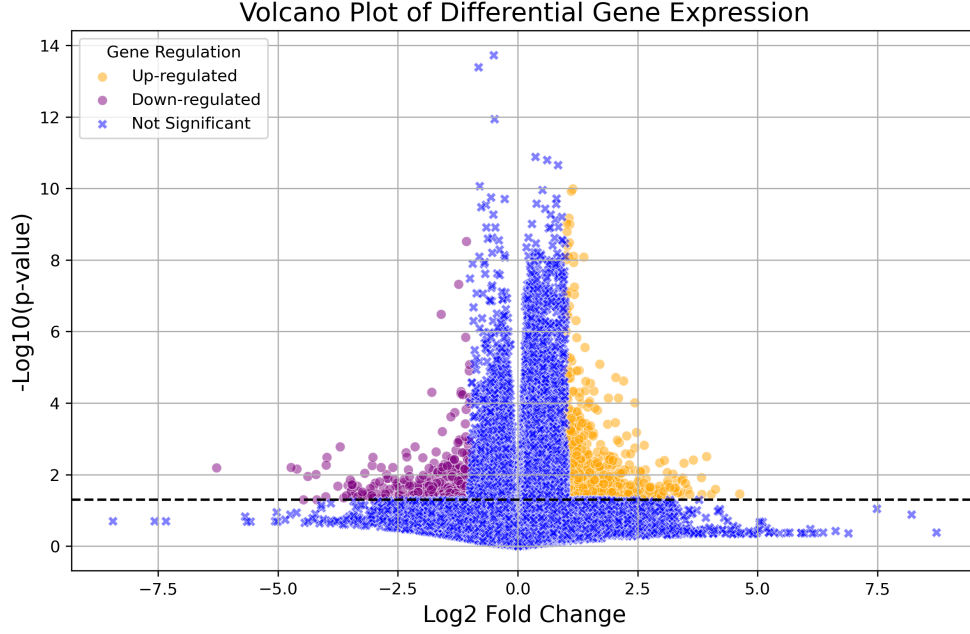


Fig. 3 This Volcano plot visualizes differential gene expression, highlighting up-regulated, down-regulated, and non-significant genes.

plot. A total of 789 genes were identified as up-regulated, while 377 genes were down-regulated. The plot effectively highlights these differences, with points color-coded to indicate their significance. The substantial log2 fold changes and corresponding negative log10 p-values suggest that the data has the potential to distinguish between early and late stages of prostate cancer [Figure 3](#).

3.3 Machine Learning Performance

In this study, we employed seven advanced supervised machine learning algorithms to classify gene expression data for distinguishing between early and late stages of cancer. The algorithms used include RF, DT, KNN, LR, NB, SVM, and XGB. The results presented are based on 100 runs of each algorithm to ensure robustness and reliability in our findings. To address the challenges posed by the imbalanced dataset, we utilized SMOTE to generate synthetic samples for the minority class, thereby enhancing model performance.

The performance metrics were evaluated using precision, recall, and F1-score, as illustrated in [Table 5](#). The results indicate that the models benefited significantly from hyperparameter tuning through grid search, which optimized the performance of each algorithm [Table 3](#).

Table 3 Selected Hyperparameters for Different algorithms

Method	Hyperparameter	Non-Normal Data
RF	n_estimators	50
	max_depth	20
	min_samples_split	10
DT	criterion	entropy
	max_depth	20
	min_samples_split	2
	min_samples_leaf	2
KNN	n_neighbors	11
	weights	uniform
	metric	manhattan
LR	C	0.001
	penalty	l2
	solver	saga
	max_iter	200
NB	var_smoothing	1e-9
SVM	C	1
	kernel	rbf
	gamma	scale
XGB	n_estimators	100
	max_depth	5
	learning_rate	0.1

3.3.1 Model evaluation

The comparative analysis of model performance metrics with and without enhancement methods reveals significant insights into the effectiveness of these techniques. When examining precision, recall, and F1-score across various algorithms, it is evident that enhancement methods markedly improve performance. The most notable improvements were observed in SVM, which showed the highest enhancements among all models, highlighting the significant benefits of feature selection and other enhancements. Specifically, SVM demonstrated remarkable gains across all performance metrics. Its average precision increased from 47.54% to 68.30%, a rise of 20.71% percentage points. Mean recall also improved, going from 61.52 to 66.30, reflecting a gain of 4.78% percentage points. Additionally, the mean F1-score saw a substantial boost, climbing from 49.81% to 66.68, resulting in an improvement of 16.87% percentage points. According to [Table 5](#), the improvements in SVM were evident in both its average performance and its best individual performance across the various metrics.

In contrast, Random Forest (RF) and Logistic Regression (LR) algorithms demonstrate the best overall performance compared to the other methods evaluated. For the mean metrics, RF achieved the highest precision at 70.75%, recall at 70.51%, and F1-score at 70.20%, indicating its ability to consistently produce strong results. Similarly, LR also performed well, with a precision of 70.80%, recall of 68.93%, and F1-score of 69.30%. Furthermore, the RF algorithm attained the best individual performance,

Table 4 Model performance metrics without feature selection highlight the mean results, derived from 100 repeated trials, and the best results across various algorithms.

		RF	DT	KNN	LR	NB	SVM	XGB
Mean	Precision	68.77	62.19	59.28	65.78	68.24	47.54	69.82
	Recall	68.76	61.66	59.76	65.20	62.61	61.52	69.85
	F1-score	66.79	61.61	58.85	65.21	63.01	49.81	69.07
The best	Precision	76.00	78.00	69.00	76.00	76.00	69.00	79.00
	Recall	77.00	72.00	68.00	76.00	72.00	71.00	79.00
	F1-score	76.00	74.00	69.00	76.00	73.00	70.00	79.00

Table 5 The comparison of model performance metrics using SelectFpr (alpha = 0.05) highlights the mean results, derived from 100 repeated trials, and the best results across various algorithms.

		RF	DT	KNN	LR	NB	SVM	XGB
	score_func	f_classif	f_classif	f_classif	f_classif	f_classif	f_classif	f_classif
Mean	Precision	70.75	62.24	66.56	70.80	69.89	68.30	70.73
	Recall	70.51	60.96	53.27	68.93	65.06	66.30	70.49
	F1-score	70.20	61.25	51.11	69.30	65.53	66.68	70.20
The best	Precision	83.00	75.00	76.00	80.00	75.00	78.00	79.00
	Recall	83.00	72.00	65.00	80.00	73.00	79.00	80.00
	F1-score	83.00	73.00	65.00	80.00	74.00	79.00	79.00

with the highest precision, recall, and F1-score of 83% across all three metrics in the performance matrix. This balanced and superior performance of the RF algorithm across both the mean and best metrics makes it the standout choice among the algorithms presented in the analysis. Given its strong performance, the Random Forest algorithm appears to be an appropriate method for detecting both early and late stage cancer.

Furthermore, The analysis of the boxplots generated from the 100-run results reveals that the use of feature selection consistently improves the mean performance across the different models, as indicated by the higher median values. Additionally, the confidence interval, represented by the boxplot height, is reduced when FS is applied, suggesting more reliable and stable results with a narrower spread of the data [Figure 4](#).

3.3.2 Cross Validation

The results of 10-fold cross-validation experiments demonstrate a significant improvement in performance across all the evaluated machine learning algorithms. SVM model showed the most substantial enhancement, with the mean score increasing from 50.35 to 69.63 when utilizing proposed feature selection technique. Furthermore, RF and XGB models exhibited the strongest overall performance, with mean scores exceeding 80% using feature selection approach. DT, KNN, LR, and NE models also saw improvements in their cross-validation results, though the gains were more modest

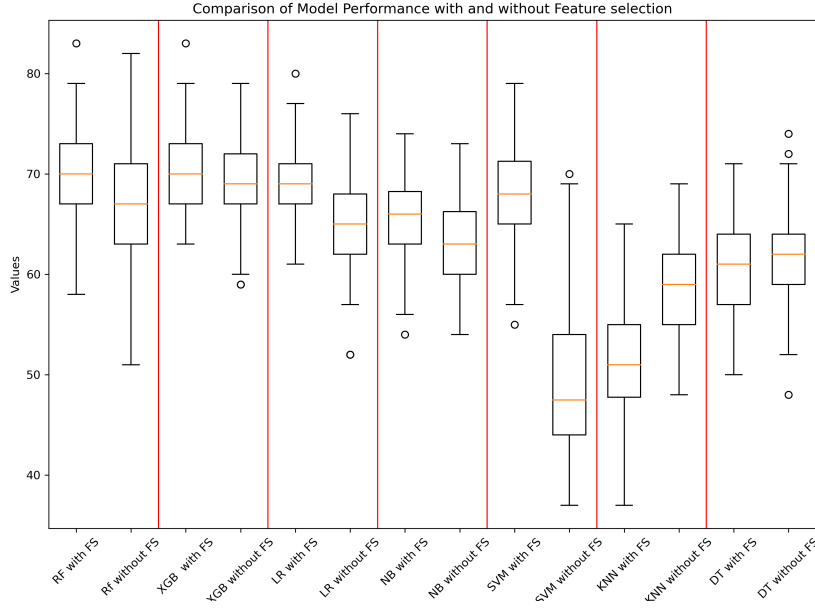


Fig. 4 Evaluation of test results for model performance with and without feature selection.

compared to the top-performing algorithms [Table 6](#). All 100 outcomes from the 10-fold cross-validation are displayed in boxplots. The findings indicate that both RF and XGB achieve the highest F1-weighted scores when feature selection is applied. In contrast, KNN demonstrates the lowest performance among the models presented [Figure 5](#).

Table 6 10-fold cross validation results

	result	RF	DT	KNN	LR	NB	SVM	XGB
Without F_S	Mean	66.41	61.12	57.96	64.52	63.54	50.35	68.27
	Best	63.07	63.09	53.33	59.99	60.07	55.09	66.13
With F-S	Mean	80.29	69.84	58.96	73.87	76.81	69.63	80.24
	Best	80.36	70.10	60.26	74.63	76.89	72.08	79.73

3.4 Deep Learning

To evaluate the impact of different preprocessing strategies on the neural network model's performance, we ran experiments across various configurations, focusing on normalization, dimensionality reduction, and data augmentation. Below, we present the results and insights from these experiments.

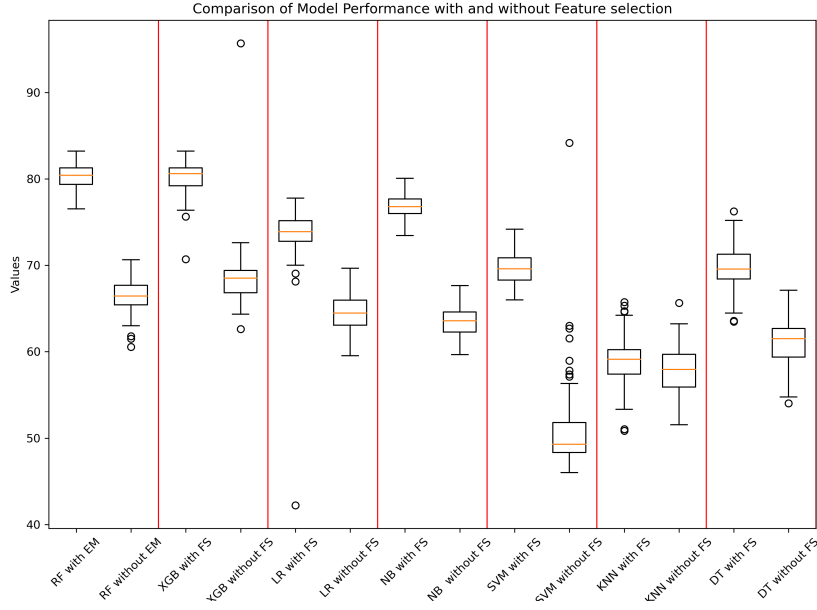


Fig. 5 Assessment of model performance with and without feature selection utilizing 10-fold cross-validation.

3.4.1 Neural Network Model

a sequential neural network model comprising five dense layers was employed to address the research objectives. The architecture begins with an input layer that outputs 256 units, followed by layers that progressively reduce the dimensionality to 128, 64, 32, and culminates in a single output unit. This model contains a total of 15,572,225 trainable parameters, enabling it to capture complex patterns within the data. While the high parameter count enhances the model's capacity for learning intricate relationships, it also necessitates careful management to mitigate the risk of overfitting [Table 7](#).

Table 7 Summary of the model architecture

Layer (type)	Output Shape	Param
dense (Dense)	(None, 256)	15,528,960
dense_1 (Dense)	(None, 128)	32,896
dense_2 (Dense)	(None, 64)	8,256
dense_3 (Dense)	(None, 32)	2,080
dense_4 (Dense)	(None, 1)	33

Total params: 15,572,225 (59.40 MB)

Trainable params: 15,572,225 (59.40 MB)

Non-trainable params: 0 (0.00 B)

3.4.2 Baseline Model (Normalization Only, No Dimensionality Reduction or Augmentation)

The baseline model was trained with only normalization applied to the original 486 samples, each with 60, 660 gene features. This model yielded an accuracy of 67.12%, which serves as a control for comparison with other preprocessing strategies.

The relatively modest performance in the baseline scenario highlights the challenges of training on high-dimensional data with limited sample size. With a large feature set, the model may struggle to capture meaningful patterns without overfitting, which is a common issue in genomic and other biological datasets.

3.4.3 Dimensionality Reduction

Dimensionality reduction was applied in two forms: Principal Component Analysis (PCA) and Independent Component Analysis (ICA). Both approaches aimed to mitigate the risk of overfitting by reducing the number of input features, which should theoretically enhance the model’s ability to generalize on the test and validation sets.

PCA Dimensionality Reduction : After applying PCA, the dimensionality of the input data was reduced from 60, 660 to 100 principal components. These 100 components were selected to retain the maximum variance from the original data, effectively compressing information while removing noise. Training the neural network with PCA-reduced data resulted in an accuracy of 69.86%.

This improvement over the baseline suggests that PCA’s ability to retain the most significant components of variance allowed the model to learn more relevant patterns, indicating PCA’s utility in reducing noise while preserving core features. However, there may still be some information loss that prevents even higher accuracy.

ICA Dimensionality Reduction : Similarly, ICA was used to reduce the dataset’s dimensionality to 100 independent components. Unlike PCA, which maximizes variance, ICA seeks to identify statistically independent components within the data. When trained on this ICA-reduced dataset, the model achieved an accuracy of 60.27%, lower than both the baseline and the PCA result.

The reduced performance with ICA could be due to the nature of the gene expression data, where the most informative features may be related by variance rather than independence. ICA’s emphasis on statistical independence may result in the extraction of components less aligned with the biological patterns, thus leading to information loss.

3.4.4 Data Augmentation

Data augmentation was implemented to address the limited number of samples by synthetically expanding the training dataset. This approach involved adding small amounts of Gaussian noise to each data point, producing a 10x larger training set. This augmentation technique was tested both with the full 60, 660-dimensional data and with PCA- and ICA-reduced datasets.

Augmentation Without Dimensionality Reduction : When training was conducted on the full 60, 660-dimensional dataset with 10x augmentation, the model

achieved an accuracy of 71.23%, the highest among all scenarios tested. This outcome demonstrates the advantage of synthetic data in enhancing the model’s robustness by preventing overfitting on the original 486 samples.

In high-dimensional settings, where biological noise can vary, Gaussian noise augmentation helps the model generalize better by exposing it to slight variations, simulating biological variability. This result suggests that augmentation is particularly valuable for this dataset, as it allows the model to learn more consistent patterns across a broader range of inputs.

Augmentation with PCA Reduction : Applying data augmentation to the PCA-reduced dataset (reduced to 100 components) yielded a final accuracy of 65.75%, which is lower than the PCA-only model. This suggests that, while augmentation benefits the full-dimensional model, its effectiveness may diminish when combined with dimensionality reduction. It’s possible that adding noise to already compressed data may distort the core information retained after PCA, leading to a slight performance decrease.

Augmentation with ICA Reduction : Finally, training with data augmentation on the ICA-reduced dataset resulted in an accuracy of 69.86%, a significant improvement over ICA without augmentation. In this scenario, augmentation compensates for some of the limitations posed by ICA’s extraction of independent components. The noise introduced may help the model to generalize across potential variations, partially offsetting the limitations of ICA by adding diversity back into the dataset.

3.4.5 comparison

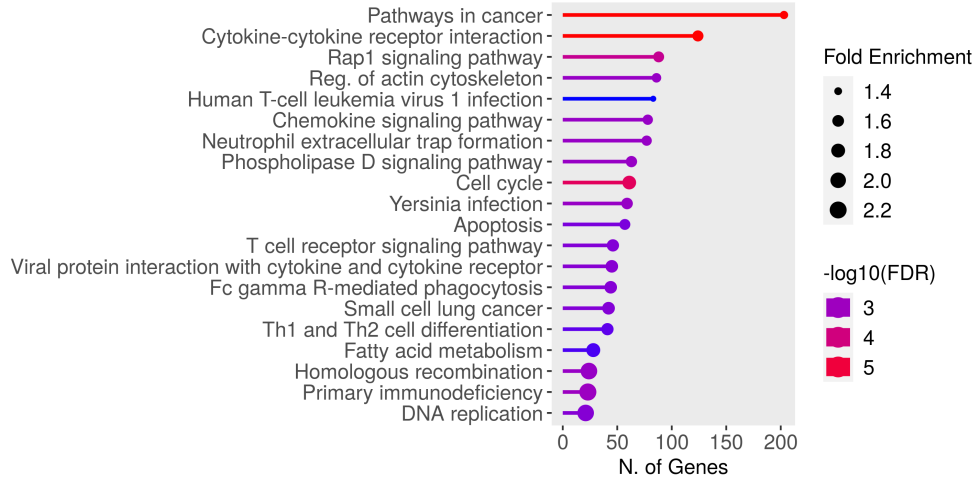
These results illustrate that:

1. PCA without augmentation provides a modest increase in accuracy, suggesting PCA is effective for variance-focused feature reduction.
2. ICA without augmentation may not be as effective for this dataset due to potential information loss in independent components, which could miss variance-related patterns.
3. Data augmentation on the full-dimensional dataset yields the best performance, showing its value in counteracting overfitting.
4. Augmentation combined with PCA shows some performance decline, likely due to over-compression.
5. Augmentation with ICA improves performance closer to PCA-only results, highlighting augmentation’s role in enhancing generalization even with independently reduced data.

These findings indicate that while dimensionality reduction and augmentation both aim to address limited sample size and high feature count, they interact differently depending on the reduction method. Data augmentation alone had the most significant impact, making it a potentially powerful approach in similar high-dimensional biological datasets [Table 8](#).

Table 8 Deep Learning Model Evaluation

Model	Normalization	Augmentation	Reduction	Accuracy
Baseline	✓	X	X	67.12
PCA Reduction	✓	X	✓	69.86
ICA Reduction	✓	X	✓	60.27
Augmentation Only	X	✓	X	71.23
Augmentation + PCA Reduction	✓	✓	✓	65.75
Augmentation + ICA Reduction	✓	✓	✓	69.86

**Fig. 6** GO Enrichment analysis result for selected genes

3.5 GO Enrichment Analysis

The pathway enrichment analysis performed using the ShinyGO tool provides valuable insights into biological functions and is based on the optimal classification results from the Random Forest model, which achieved an accuracy of 83%. The enriched pathways are organized by their gene counts [12]. Enriched pathways are organized based on their gene numbers. The most significant pathways are marked in red, while those with lesser significance are shown in blue, corresponding to their $\log_{10}(\text{FDR})$ values. Larger dots in the graph indicate a greater number of fold enrichment values associated with each pathway. The pathway with the largest number of genes represented is "Pathways in cancer", which indicates that the selection of genes in this study was appropriate and effective at capturing the key genes involved in cancer-related pathways.

However, the pathway with the largest dot size, suggesting the greatest fold enrichment values, is "Primary immunodeficiency". This means that the genes involved in the "Primary immunodeficiency" pathway show the strongest enrichment or overrepresentation compared to the background, even though the total number of genes in this pathway is not the highest [Figure 6](#).

4 Conclusion

This research underscores the transformative potential of machine learning and advanced feature selection techniques in enhancing the pathological staging predictions of prostate cancer. By leveraging RNA sequencing data from The Cancer Genome Atlas (TCGA), we successfully differentiated between early and late stages of prostate cancer, demonstrating the critical role of accurate staging in guiding treatment decisions and improving patient outcomes.

The findings underscore the effectiveness of the Random Forest test result algorithm, which was recognized as the most efficient model, achieving high precision, recall, and F1-scores of up to 83% and a maximum cross-validation score of 80.38%. This reliability, along with significant performance enhancements attained through hyperparameter tuning and feature selection, highlights the importance of these methodologies in clinical applications.

Moreover, the exploration of data augmentation and feature extraction techniques such as PCA, revealed their essential contributions to model interpretability and performance. These approaches not only streamline data processing but also address the challenges posed by high-dimensional gene expression data, facilitating more reliable diagnostic tools.

Overall, this research presents a significant advancement in the accurate staging of prostate cancer, enabling the determination of the appropriate level of cancer aggressiveness. By ensuring that patients will receive the right amount and dosage of medication, this approach not only optimizes treatment efficacy but also reduces the duration of therapy and alleviates the financial and temporal burdens on patients. These findings underscore the potential for enhanced patient outcomes through precise staging methodologies. Future work should focus on validating these models in pathological settings and exploring additional omics data to further refine staging predictions and improve therapeutic strategies for prostate cancer patients.

Acknowledgements

We acknowledge TCGA (The Cancer Genome Atlas) for its free use.

Author contributions

R.G and M.G: Conceptualization, visualization, Conducting data collection and analysis. R.G, M.G and MA.A: methodology and writing-original draft preparation. MA.A, SM.A and A.D: Offering critical feedback on data interpretation and analysis, Reviewing and editing the final draft.

Funding

This study did not receive any funding.

Data availability

The datasets for this study were collected from The Cancer Genome Atlas (TCGA) (<https://www.cancer.gov/tcga>), a publicly available repository for cancer analysis.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent to participate

Not applicable.

Competing interests

The authors declare no competing interests.

References

- [1] Sheena Angra and Sachin Ahuja. Machine learning and its applications: A review. In *2017 international conference on big data analytics and computational intelligence (ICBDAC)*, pages 57–60. IEEE, 2017.
- [2] Muhammad Arham Tariq. A study on comparative analysis of feature selection algorithms for students grades prediction. *Journal of Information and Organizational Sciences*, 48(1):133–147, 2024.
- [3] Neil Bannister and John Broggio. Cancer survival by stage at diagnosis for england (experimental statistics): adults diagnosed 2012, 2013 and 2014 and followed up to 2015. *Produced in collaboration with Public Health England*, 2016.
- [4] Erkan Bostanci, Engin Kocak, Metehan Unal, Mehmet Serdar Guzel, Koray Acici, and Tunc Asuroglu. Machine learning analysis of rna-seq data for diagnostic and prognostic prediction of colon cancer. *Sensors*, 23(6):3080, 2023.
- [5] Freddie Bray, Mathieu Laversanne, Hyuna Sung, Jacques Ferlay, Rebecca L Siegel, Isabelle Soerjomataram, and Ahmedin Jemal. Global cancer statistics 2022: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 74(3):229–263, 2024.
- [6] Francois Chollet. *Deep learning with Python*. Simon and Schuster, 2021.
- [7] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.
- [8] Xiaofeng Dai and Li Shen. Advances and trends in omics technology development. *Frontiers in Medicine*, 9:911861, 2022.
- [9] Padideh Danaee, Reza Ghaeini, and David A Hendrix. A deep learning approach for cancer detection and relevant gene identification. In *Pacific symposium on biocomputing. pacific symposium on biocomputing*, volume 22, page 219. NIH Public

Access, 2016.

- [10] Thomas W Edgar and David O Manz. *Research methods for cyber security*. Syngress, 2017.
- [11] Giorgio Gandaglia, Riccardo Leni, Freddie Bray, Neil Fleshner, Stephen J Freedland, Adam Kibel, Pär Stattin, Hendrick Van Poppel, and Carlo La Vecchia. Epidemiology and prevention of prostate cancer. *European urology oncology*, 4(6):877–892, 2021.
- [12] Steven Xijin Ge, Dongmin Jung, and Runan Yao. Shinygo: a graphical gene-set enrichment tool for animals and plants. *Bioinformatics*, 36(8):2628–2629, 2020.
- [13] Osama Hamzeh, Abedalrhman Alkhateeb, Julia Zheng, Srinath Kandalam, and Luis Rueda. Prediction of tumor location in prostate cancer tissue using a machine learning system on gene expression data. *BMC bioinformatics*, 21:1–10, 2020.
- [14] CJ Harvey, J Pilcher, Jonathan Richenberg, Uday Patel, and Ferdinand Frauscher. Applications of transrectal ultrasound in prostate cancer. *The British journal of radiology*, 85(specialIssue.1):S3–S17, 2012.
- [15] Darius Jегelevičius, Arūnas Lukoševičius, Alvydas Paunksnis, and Valerijus Barzdžiukas. Application of data mining technique for diagnosis of posterior uveal melanoma. *Informatica*, 13(4):455–464, 2002.
- [16] Hugh J Jewett. Significance of the palpable prostatic nodule. *Journal of the American Medical Association*, 160(10):838–839, 1956.
- [17] Fei Jiang, Yong Jiang, Hui Zhi, Yi Dong, Hao Li, Sufeng Ma, Yilong Wang, Qiang Dong, Haipeng Shen, and Yongjun Wang. Artificial intelligence in healthcare: past, present and future. *Stroke and vascular neurology*, 2(4), 2017.
- [18] Asad Khattak, Muhammad Zubair Asghar, Zain Ishaq, Waqas Haider Bangyal, and Ibrahim A Hameed. Enhanced concept-level sentiment analysis system with expanded ontological relations for efficient classification of user reviews. *Egyptian Informatics Journal*, 22(4):455–471, 2021.
- [19] Michał Koziarski, Bogusław Cyganek, Przemysław Niedziela, Bogusław Olborski, Zbigniew Antosz, Marcin Żydak, Bogdan Kwolek, Paweł Wasowicz, Andrzej Buła, Jakub Swadźba, et al. Diagset: a dataset for prostate cancer histopathological image classification. *Scientific Reports*, 14(1):6780, 2024.
- [20] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [21] Tae-Hwy Lee, Aman Ullah, and Ran Wang. Bootstrap aggregating and random forest. *Macroeconomic forecasting in the era of big data: Theory and practice*,

pages 389–429, 2020.

- [22] Andrzej Maćkiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers & Geosciences*, 19(3):303–342, 1993.
- [23] Sherif Mehralivand, Dong Yang, Stephanie A Harmon, Daguang Xu, Ziyue Xu, Holger Roth, Samira Masoudi, Deepak Kesani, Nathan Lay, Maria J Merino, et al. Deep learning-based artificial intelligence for prostate cancer detection at biparametric mri. *Abdominal Radiology*, 47(4):1425–1434, 2022.
- [24] Rory Mitchell and Eibe Frank. Accelerating the xgboost algorithm using gpu computing. *PeerJ Computer Science*, 3:e127, 2017.
- [25] Alan W Partin, John Yoo, H Ballentine Carter, Jay D Pearson, Daniel W Chan, Jonathan I Epstein, and Patrick C Walsh. The use of prostate specific antigen, clinical stage and gleason score to predict pathological stage in men with localized prostate cancer. *The Journal of urology*, 150(1):110–114, 1993.
- [26] Olivier Regnier-Coudert, John McCall, Robert Lothian, Thomas Lam, Sam McClinton, and James N’Dow. Machine learning for improved pathological staging of prostate cancer: a performance comparison on a range of classifiers. *Artificial intelligence in medicine*, 55(1):25–35, 2012.
- [27] EMIEL TH RUIJTER, CHRISTINA A VAN DE KAA, JACK A SCHALKEN, FRANS M DEBRUYNE, and DIRK J RUITER. Histological grade heterogeneity in multifocal prostate cancer. biological and clinical implications. *The Journal of pathology*, 180(3):295–299, 1996.
- [28] Siva Singireddy, Abed Alkhateeb, Iman Rezaeian, Luis Rueda, Dora Cavallo-Medved, and Lisa Porter. Identifying differentially expressed transcripts associated with prostate cancer progression using rna-seq and machine learning techniques. In *2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–5. IEEE, 2015.
- [29] Octavian Sabin Tătaru, Mihai Dorin Vartolomei, Jens J Rassweiler, Oşan Virgil, Giuseppe Lucarelli, Francesco Porpiglia, Daniele Amparore, Matteo Manfredi, Giuseppe Carrieri, Ugo Falagario, et al. Artificial intelligence and machine learning in prostate cancer patient management—current trends and future perspectives. *Diagnostics*, 11(2):354, 2021.
- [30] AE Tercan and B Sohrabian. Multivariate geostatistical simulation of coal quality data by independent components. *International Journal of Coal Geology*, 112:53–66, 2013.
- [31] Shahadat Uddin, Sirui Yan, and Haohui Lu. Machine learning and deep learning in project analytics: methods, applications and research trends. *Production Planning & Control*, pages 1–20, 2024.

[32] Harry Zhang. The optimality of naive bayes. *Aa*, 1(2):3, 2004.