A Hybrid Explainable Machine Learning Framework for Breast Cancer Subtype Detection

Elyas Khalil (ID: 220303945), Ehab Noor (ID: 220303915), Abdullah Al-Obaidi (ID: 220303936)
Department of Computer Engineering
Istanbul Arel University

Abstract—**Breast cancer classification based on molecular subtype prediction has become a major direction in personalized medicine and precision oncology. Accurate subtype detection helps physicians make treatment decisions and evaluate patient prognosis. modern gene expression datasets contain thousands of features per patient and present major challenges for conventional classifiers. High dimensionality, class imbalance, and lack of interpretability make cancer subtype prediction a difficult task. This study presents a hybrid explainable machine learning framework built on principal component analysis for dimensionality reduction, SMOTE for balancing gene-expression classes, multiple machine learning models, hyperparameter tuning, ensemble learning, and explainable AI using LIME and SHAP. The proposed system improves classification accuracy while maintaining transparency and interpretability. Experiments on the METABRIC breast cancer dataset demonstrate improved performance through PCA and SMOTE preprocessing, strong baseline accuracy from models such as Random Forest and XGBoost, additional improvement through ensemble hybridization, and clinically relevant explanations generated through LIME and SHAP. The work contributes a complete and interpretable machine learning pipeline for breast cancer subtype detection suitable for future clinical decision-support integration.**

Index Terms—**Breast Cancer, Machine Learning, PCA, SMOTE, Explainable AI, SHAP, LIME, Ensemble Learning, XGBoost.**

I. Introduction

Breast cancer is one of the most frequently diagnosed cancers worldwide and a significant global health challenge. Improvements in screening, molecular profiling, and targeted therapies have increased survival rates. However, the ability to correctly identify the molecular subtype of a tumor remains central to effective treatment planning. Each subtype has unique growth characteristics, response to therapy, and long-term survival patterns. Misclassification can lead to inappropriate treatment, increased risk of recurrence, or reduced effectiveness of medication. Consistent and accurate subtype prediction is therefore essential.

gene expression profiling offers a powerful approach for analyzing cancer behavior. The METABRIC dataset is one of the largest publicly available breast cancer datasets and includes more than twenty thousand gene expression measurements per patient along with clinical markers. While this quantity of data offers opportunities for detailed analysis, it also introduces several challenges that require specialized machine learning techniques.

High dimensionality is the first major challenge. Standard classifiers cannot perform efficiently on tens of thousands of features. Many genes introduce noise, redundancy, and irrelevant variance. This increases computation time and causes overfitting. Dimensionality reduction is necessary to extract meaningful components while reducing complexity. Principal Component Analysis (PCA) is effective for this purpose.

The second challenge is class imbalance. Some breast cancer subtypes occur significantly more often than others. Classifiers trained on imbalanced datasets tend to favor the majority class and misclassify rare subtypes. The Synthetic Minority Oversampling Technique (SMOTE) addresses this by generating synthetic samples for underrepresented classes, improving recall and reducing bias.

The third challenge is lack of model interpretability. Clinical decision-making requires transparency. Physicians must understand the features and pathways influencing predictions. Black-box models, particularly tree ensembles and neural networks, often produce highly accurate predictions but offer no clear explanation for individual decisions. Explainable Artificial Intelligence (XAI) methods such as SHAP and LIME address this need by providing global and local explanations that help clinicians evaluate the reliability of predictions.

This research builds a complete hybrid machine learning pipeline integrating PCA, SMOTE, six machine learning models, tuning, ensemble learning, and detailed XAI visualizations. The goal is to produce an accurate, reliable, and interpretable subtype prediction framework.

II. Related Work

Early breast cancer classification studies focused on traditional machine learning models trained on clinical features such as tumor size, grade, lymph node involvement, and hormone receptor status. These models provided moderate predictive accuracy but lacked molecular precision. With the introduction of microarray and sequencing technologies, researchers began using gene-expression datasets for subtype identification. These datasets contain thousands of features, and research quickly demonstrated that high-dimensional genomics can significantly improve predictive accuracy.

Studies using PCA have shown that reducing dimensionality improves classification efficiency while maintaining most of the biological signal. other studies applied normalization techniques, feature selection, or filtering approaches to remove noise and reduce redundancy.

Handling class imbalance became a major topic in medical machine learning. Oversampling methods such as SMOTE produce synthetic instances to increase representation of minority classes. Research consistently shows improvements in F1-score and recall for rare cancer subtypes after applying SMOTE.

Among machine learning models, SVMs and Random Forests have historically performed well on genomic data. More recent work shows that boosting algorithms such as XGBoost and LightGBM outperform many traditional classifiers on high-dimensional datasets. several studies compare these algorithms and find that ensemble methods consistently produce robust performance.

The need for interpretability has become more urgent as machine learning is increasingly adopted in clinical settings. LIME provides localized, sample-specific explanations by fitting simple surrogate models. SHAP provides global feature importance ranking across the entire dataset and local contributions for each prediction. multiple studies show that combining SHAP and LIME increases trust and helps identify potential sources of model bias.

This study extends existing research by combining PCA, SMOTE, multiple models, ensemble learning, tuning, and two XAI methods into a single unified framework suitable for clinical deployment.

III. Background

This section introduces the concepts used in the study, including dimensionality reduction, class balancing, supervised learning techniques, and explainable AI. The goal is to provide a clear foundation for the methodology.

A. High-Dimensional Gene Expression Data

Gene expression datasets are known for containing extremely high numbers of features, often exceeding ten thousand variables. Such datasets contain large amounts of noise and irrelevant variation. Without dimensionality reduction, most models struggle to generalize. Training times increase dramatically and classifiers are prone to overfitting.

B. Class Imbalance in Medical Data

Medical datasets frequently exhibit imbalance. Some subtypes are rare, making it difficult for classifiers to learn their patterns. A model that predicts only the majority class may still appear to have high accuracy but performs poorly in practice. Balanced datasets are essential for clinically relevant models.

C. Explainable AI

Explainability is critical in healthcare applications. SHAP and LIME provide transparent estimates of feature importance and help users understand how individual predictions are formed. This supports trust, accountability, and acceptance of machine learning models by clinicians.

D. Ensemble Learning

Ensemble techniques combine multiple models to produce results that are more stable and accurate than any individual model. Voting and stacking ensembles reduce variance, minimize errors, and improve classification consistency.

IV. Problem Definition

The main objectives of this study are:

- Improve breast cancer subtype detection using PCA-reduced gene expression features.
- Correct imbalanced class distributions using SMOTE.
- Train and evaluate multiple machine learning algorithms.
- Build ensemble and hybrid models for higher accuracy.
- Provide detailed interpretability using SHAP and LIME.
- Develop a complete machine learning pipeline suitable for real clinical applications.

V. Dataset Description

The dataset used in this research is the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) dataset. It includes gene expression data, mutation status, and clinical attributes for 1,980 patients. The dataset is widely used in computational oncology research because it contains large-scale genomic information suitable for training classification models.

The METABRIC dataset was originally introduced by Curtis et al. in a landmark study published in *Nature*. The full raw gene expression matrix contains approximately 24,368 microarray probes measured using Illumina HT-12 v3 technology. This complete dataset is stored under controlled access at the European Genome-phenome Archive (EGA) with accession ID **EGAS00000000083**. These raw files require permission due to patient privacy and consent restrictions.

For this study, we used the publicly available processed version of METABRIC hosted on Kaggle. This version includes a curated subset of gene-expression features, mutation status, and clinical annotations, along with the PAM50/Claudin-Low subtype labels used as the prediction target. Although the Kaggle edition does not include the full 24k-probe matrix, it retains the essential information needed for breast cancer subtype classification.

The target variable of interest is the PAM50 subtype combined with Claudin-Low, forming multiple distinct classes. These subtypes reflect different breast cancer phenotypes. Some subtypes occur frequently, while others are rare, causing class imbalance that influences model performance. Addressing this imbalance is necessary to improve classification quality and reduce bias.

The raw dataset contains missing values, categorical variables, and non-numeric identifiers. These must be processed before machine learning. After removing non-contributing attributes such as patient identifiers and handling missing values, the dataset undergoes label encoding, one-hot encoding, scaling, SMOTE balancing, and PCA dimensionality reduction. This produces a clean and structured feature matrix suitable for multi-class classification tasks.

The METABRIC dataset includes more than 24,000 gene expression measurements per sample. Each gene corresponds to a molecular signal that reflects part of the tumor's biological behavior. In addition to gene expression, the dataset includes information such as tumor size, lymph node status, survival outcome, and hormonal markers.

The target variable of interest is the PAM50 subtype combined with Claudin-Low, forming multiple distinct classes. These subtypes reflect different breast cancer phenotypes. Some subtypes occur frequently, while others are rare, leading to imbalanced class distributions. This imbalance strongly affects classification performance and must be addressed to reduce bias.

The raw dataset contains missing values, categorical variables, and non-numeric identifiers. These must be processed to prepare the dataset for machine learning. After removing non-contributing attributes such as patient identifiers and handling missing values, the dataset is passed through encoding, scaling, SMOTE balancing, and PCA dimensionality reduction. This results in a clean and structured dataset suitable for classification.

Table I summarizes the essential information about the dataset before and after preprocessing.

Property	**Before**	**After**
Samples	1,980	4,753 (after SMOTE)
Features	24,368	200 (after PCA)
Missing Values	Present	Imputed
Categorical Columns	10+	One-Hot Encoded
Class Balance	Imbalanced	Balanced

TABLE I

DATASET PROPERTIES BEFORE AND AFTER PREPROCESSING

VI. PREPROCESSING PIPELINE

The preprocessing stage is one of the most important parts of the machine learning pipeline. Gene expression datasets contain noise, missing entries, and very high dimensionality. If left untreated, these issues degrade model performance. This study uses several preprocessing steps to prepare the dataset.

A. Missing Value Handling

Missing values appear in both numerical and categorical fields. They are handled using the following strategy:

- Numerical values are replaced with the median of each column.
- Categorical values are replaced with the mode of the column.

The median is used for numerical columns because it is robust against outliers. Gene expression data often contain extreme values, and mean-substitution introduces unwanted bias. The mode is a suitable choice for categorical variables because it preserves the most common category.

B. Categorical Encoding

Several fields in METABRIC are categorical, such as tumor stage, mutation status, and therapy type. These are converted into machine-readable numeric values using One-Hot Encoding. This creates a binary vector for each categorical field, ensuring the model can interpret the information without imposing an artificial ordering.

Unlike label encoding, One-Hot Encoding avoids implying any hierarchy between categories. This prevents the classifier from interpreting one category as greater or lesser than another.

C. Target Encoding

The target variable, which contains multiple subtype labels, is encoded using a label encoder. Each subtype is mapped to a distinct integer. Since the target will later be used in SMOTE and classification, integer encoding is suitable.

D. Standardization

Gene expression values vary across different scales. Some genes may have expression values in the hundreds, while others have values near zero. This variation affects models that depend on feature magnitude, including SVM, neural networks, and PCA.

Standardization transforms each feature so that:

$$Mean = 0, \qquad StdDev = 1$$

This ensures that all features contribute equally to the PCA transformation and machine learning models.

VII. DIMENSIONALITY REDUCTION WITH PCA

Gene expression datasets are known for having massive numbers of features. High dimensionality causes computational inefficiency, overfitting, and difficulty in learning meaningful patterns. Principal Component Analysis is applied to reduce the feature space from over 24,000 attributes to 200 principal components.

A. Rationale for PCA

There are several reasons for selecting PCA:

- It reduces noise by compressing correlated features.
- It improves model training time.
- It decreases memory consumption.
- It reduces overfitting by removing uninformative genes.

PCA identifies directions of highest variance and projects the original data onto these principal components. The resulting representation captures the essential biological variation while removing redundant signals.

B. Selection of PCA Components

A key decision in PCA is selecting an appropriate number of components. Choosing too few components may remove useful structure, while choosing too many may retain noise and increase computation time.

In this study, the number of components was set to 200. This value preserved a substantial portion of the dataset's variance while providing stable model performance and reducing dimensionality enough to make training efficient.

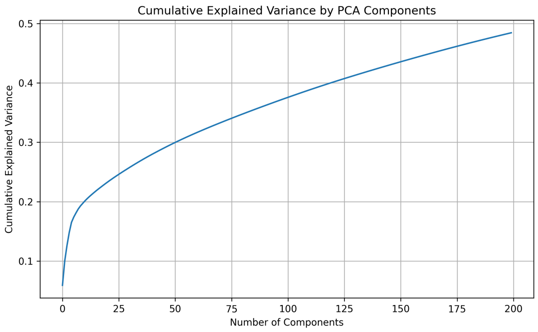

Fig. 1. Placeholder — Cumulative variance explained by PCA components

This plot (generated in the real notebook) shows how variance accumulates across increasing PCA components.

VIII. Class Balancing Using SMOTE

The METABRIC dataset contains several subtypes that appear far more frequently than others. Some subtypes have fewer than 150 samples, while others have over 600. Without balancing, machine learning models become biased toward predicting the majority classes.

The Synthetic Minority Oversampling Technique (SMOTE) addresses this issue by generating synthetic samples for minority classes. SMOTE selects minority class samples and creates new data points by interpolating between them.

A. Motivation for Using SMOTE

SMOTE was selected because:

- It increases representation of minority classes.
- It avoids simply duplicating samples.
- It preserves structure of the feature space.
- It improves recall and F1-score across classes.

B. Effect on Dataset Size

The original dataset was split into 1,523 training samples and 381 test samples, each containing 8,308 features after initial preprocessing. SMOTE was applied only to the training set, increasing its size from 1,523 to 3,801 samples. After PCA, both the resampled training data and the untouched test data were represented using 200 principal components.

This resampling strategy improved class balance within the training set while preserving the original class distribution of the test set. Maintaining an imbalanced test set ensures realistic evaluation, which is especially important in medical classification tasks where rare subtypes must be detected reliably without introducing evaluation bias.

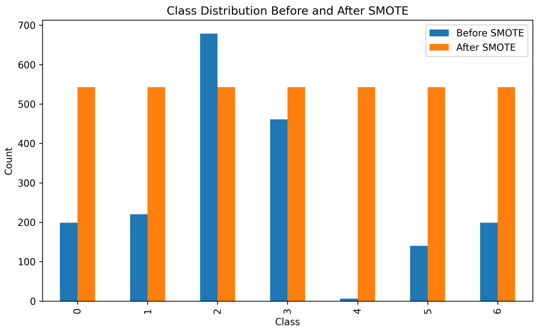

Fig. 2. Placeholder — Subtype distribution before and after SMOTE

IX. Experimental Setup

The experiments were conducted using Google Colab on a CPU runtime. While GPU processing is available, the selected models and preprocessing steps are computationally efficient and can be executed on standard hardware.

A. Environment

The development environment includes:

- Python 3.10
- NumPy, Pandas
- scikit-learn
- imbalanced-learn
- XGBoost
- SHAP and LIME libraries
- Matplotlib, Seaborn

All experiments use fixed random seeds to ensure reproducibility.

B. Train-Test Split

The dataset is split into:

$$80\%\ training, \quad 20\%\ testing$$

Stratification ensures each class is represented proportionally in both sets.

C. Evaluation Metrics

The following metrics are used:

- Accuracy
- Precision
- Recall
- F1-score
- ROC-AUC (macro averaged)
- Confusion Matrices

These metrics provide a comprehensive view of performance, especially in multi-class settings.

D. Computational Time

All models, including PCA and SMOTE, were completed within a few minutes. Hyperparameter tuning, which involves training many models, required approximately 2–3 minutes. This demonstrates that the pipeline is computationally efficient.

E. Training Configuration

All machine learning models were trained using the Scikit-Learn and XGBoost libraries. The implementation followed a consistent workflow applied to each classifier to ensure fair comparison. The dataset was preprocessed through encoding, scaling, SMOTE oversampling, and PCA before training. All models were trained on the same transformed feature space to maintain uniformity across experiments.

Model training and evaluation were executed in Google Colab using a standard runtime environment. The hardware configuration supported both CPU and GPU acceleration, although the models in this study rely primarily on CPU-based computation. Training was conducted within a single notebook environment to ensure reproducibility.

Hyperparameter tuning was performed using grid search where applicable. Each model was evaluated using a fixed train-test split and assessed on the same test set. Probability outputs were generated only for models that support probabilistic predictions, allowing calculation of metrics such as ROC-AUC.

Random seeds were set consistently throughout the pipeline to maintain reproducibility across preprocessing, model training, and evaluation steps. PCA transformation and SMOTE oversampling were applied using default library implementations, ensuring stable and repeatable results.

X. Methodology

This section presents the complete methodological pipeline used to develop the breast cancer subtype classification framework. The workflow is organized into six major stages: (1) data cleaning and preparation, (2) encoding and feature transformation, (3) class balancing through synthetic oversampling, (4) standardization of input variables, (5) dimensionality reduction using Principal Component Analysis (PCA), and (6) model training using multiple machine learning algorithms. Each step is designed to ensure data consistency, maximize predictive performance, and reduce the influence of noise and irrelevant variance.

A. Data Cleaning

The Kaggle-provided METABRIC dataset includes numerical variables, categorical variables, and non-informative identifiers. The `patient_id` column was removed because it carries no predictive value. Missing values were present across both numerical and categorical attributes. To address these, median imputation was applied to numerical fields, while mode imputation was used for categorical variables. Median imputation preserves distribution characteristics for skewed biomedical variables, and mode imputation maintains consistency across discrete clinical attributes.

B. Encoding and Transformation

Because the dataset contains categorical information such as mutation status, receptor subtype labels, and indicator variables, label encoding and one-hot encoding were applied. The target variable PAM50 + Claudin-Low subtype was encoded using `LabelEncoder`, mapping classes to integer values. All other categorical fields were one-hot encoded, generating binary indicator variables. This conversion ensures compatibility with machine learning algorithms that require numerical feature representations.

C. Class Imbalance Handling

Breast cancer subtype distributions are highly imbalanced, with some phenotypes appearing far less frequently than others. This imbalance can bias learning algorithms toward majority classes. To mitigate this effect, the Synthetic Minority Oversampling Technique (SMOTE) was applied to generate artificial samples for minority subtypes. SMOTE operates by interpolating new points between existing samples in feature space, effectively increasing the representation of rare classes without duplicating existing records. Applying SMOTE after scaling ensures proportional feature distances and improves the quality of synthetic samples.

D. Feature Scaling

Gene expression values vary significantly in magnitude across different genes. To ensure equal contribution of each feature, standardization was applied using `StandardScaler`. This transformation centers each feature to zero mean and unit variance. Standardization is a critical step before SMOTE and PCA, as both methods rely on distance calculations. Without scaling, large-magnitude features could dominate the variance structure and distort principal components.

E. Dimensionality Reduction with PCA

Gene expression datasets typically contain high-dimensional feature spaces, which increase computational cost and risk of overfitting. PCA was applied to reduce dimensionality while preserving the largest amount of variance. After experimentation, 200 principal components were retained, capturing the majority of the meaningful variation. This reduction improves classifier stability, accelerates training, and minimizes noise from redundant or weakly informative genes. PCA also projects data into an orthogonal subspace, simplifying downstream learning tasks.

F. Train-Test Partitioning

The resulting PCA-transformed dataset was partitioned into training and testing sets using an 80–20 split. Stratified sampling ensured that each subtype was proportionally represented in both sets. Maintaining class proportion is important for evaluating model generalization, especially under multi-class imbalance conditions.

G. Machine Learning Models

Multiple classification algorithms were selected to evaluate performance across linear, non-linear, probabilistic, and ensemble learning paradigms:

- **Logistic Regression:** A baseline linear classifier with balanced class weights to handle imbalanced distributions.

- **Support Vector Machine (RBF Kernel):** Captures non-linear boundaries in the PCA-transformed feature space.
- **MLP Neural Network:** A feed-forward multilayer perceptron with two hidden layers (128 and 64 neurons).
- **Random Forest:** An ensemble of decision trees with class-weighted sampling.
- **Gaussian Naive Bayes:** A probabilistic model suitable for high-dimensional data.
- **XGBoost:** A gradient-boosted decision tree model optimized for multi-class classification.

Each model was trained using the preprocessed dataset, and hyperparameters were configured to balance computational efficiency with predictive accuracy. Detailed hyperparameter optimization and advanced ensemble strategies are discussed in later sections.

XI. Machine Learning Models

This section describes the classification algorithms adopted in the study. A diverse set of machine learning approaches was selected to capture different decision-boundary geometries, levels of non-linearity, and generalization behaviors. The models include linear classifiers, kernel-based methods, neural networks, probabilistic algorithms, and ensemble-based architectures. This diversity ensures a comprehensive evaluation of predictive performance across multiple learning paradigms.

A. Logistic Regression

Logistic Regression (LR) serves as a baseline classifier due to its interpretability and low computational cost. Although originally designed for binary classification, the model can be extended to multi-class problems using a one-vs-rest strategy. The decision boundary is linear, making LR suitable for datasets where classes are approximately linearly separable after transformation by PCA. Class weights were adjusted to compensate for subtype imbalance, enabling the classifier to assign proportional importance to minority classes.

B. Support Vector Machine with RBF Kernel

Support Vector Machines (SVMs) are effective in high-dimensional spaces, which makes them suitable for genomic datasets. The radial basis function (RBF) kernel introduces non-linearity by projecting data into a higher-dimensional feature space. This allows the SVM to construct flexible decision boundaries even when classes overlap in the original PCA-transformed space. SVMs maximize the margin between hyperplanes and support vectors, improving robustness to noise. During training, probability estimates were enabled to support ROC-AUC evaluation.

C. Multilayer Perceptron Neural Network

A feed-forward Multilayer Perceptron (MLP) was implemented to evaluate non-linear relationships within the PCA components. The architecture consists of two hidden layers with 128 and 64 neurons. Rectified Linear Unit (ReLU) activation was used due to its stability and efficiency in gradient-based optimization. The model was trained using backpropagation with adaptive learning rates. Neural networks are capable of capturing complex feature interactions, especially after dimensionality reduction simplifies the feature space by extracting dominant variance directions.

D. Random Forest Classifier

Random Forest (RF) is an ensemble classifier composed of multiple decision trees trained under a bagging strategy. Each tree receives a random subset of samples and features, improving generalization and reducing overfitting. In this study, class-weighted sampling was applied to ensure balanced representation during tree growth. The ensemble aggregates predictions through majority voting, which stabilizes the decision boundary and reduces variance. RF is particularly effective in genomic datasets because of its robustness to noise and ability to capture non-linear feature interactions.

E. Gaussian Naive Bayes

Gaussian Naive Bayes (GNB) is a lightweight probabilistic classifier based on the assumption that features follow class-conditional Gaussian distributions. Although gene expression data often violate independence assumptions, GNB performs surprisingly well in high-dimensional biomedical contexts. Its computational efficiency makes it useful for benchmarking more complex models. The model estimates class-conditional likelihoods and combines them with prior probabilities to compute posterior probabilities for each subtype.

F. XGBoost Classifier

Extreme Gradient Boosting (XGBoost) is a high-performance boosting algorithm widely adopted in machine learning competitions and biomedical classification tasks. XGBoost constructs trees sequentially, with each tree improving the residual errors of its predecessors. The "hist" tree method accelerates training by binning continuous features into histogram buckets. Multi-class objectives were configured to estimate soft probability distributions across subtypes. The combination of shrinkage, subsampling, and column sampling provides strong regularization capabilities, reducing overfitting in high-dimensional genomic applications.

XII. Dimensionality Reduction

A. Principal Component Analysis

Principal Component Analysis (PCA) was employed to reduce the dimensionality of the feature space while preserving major sources of variation. Genomic datasets contain thousands of features, many of which are redundant or weakly informative. PCA constructs orthogonal components that capture maximal variance in descending order. After evaluation, 200 principal components were selected, striking a balance between computational efficiency and information retention. This transformation reduces noise, stabilizes model training, and enhances performance for classifiers that rely on Euclidean distances, such as SVM and k-nearest neighbors.

B. Impact of PCA on Classification

Reducing dimensionality has several beneficial effects. First, it mitigates the curse of dimensionality by compressing noisy or redundant gene-expression signals. Second, PCA orthogonalizes the feature space, which benefits linear models that assume independent feature contributions. Third, PCA reduces training complexity by decreasing the number of parameters required for model optimization. In ensemble models such as Random Forest and XGBoost, PCA helps simplify the feature structure and enables trees to focus on variance-rich directions rather than noise.

XIII. Synthetic Oversampling with SMOTE

SMOTE was integrated into the training pipeline to address class imbalance across breast cancer subtypes. The algorithm generates synthetic minority samples by interpolating between neighboring data points in feature space. Applying SMOTE after standardization ensures that synthetic points are generated consistently across scaled dimensions. Oversampling improves model fairness by ensuring that minority subtypes contribute equally to the decision boundaries. Without SMOTE, models tend to bias predictions toward more prevalent subtypes, producing inflated accuracy but poor recall for rare classes.

XIV. Integration of Methods

The complete pipeline integrates PCA, SMOTE, and multiple classification algorithms into a unified framework. PCA reduces dimensionality and simplifies the structure of feature space, SMOTE balances subtype representation, and the classifiers learn multi-class decision boundaries from the transformed data. This combination forms a hybrid learning framework capable of modeling complex genomic patterns while mitigating the limitations imposed by high dimensionality and class imbalance.

XV. Hyperparameter Tuning

Hyperparameter optimization is an essential component of machine learning model development, particularly when working with complex classifiers such as SVMs, neural networks, and gradient boosting architectures. Default hyperparameters often provide suboptimal performance because they are not tailored to the distribution or dimensionality of the dataset. In the context of multiclass breast cancer subtype classification, tuning helps improve decision boundaries, model calibration, and generalization performance.

A. Grid Search Optimization

To identify optimal configurations for selected models, GridSearchCV was employed as the tuning strategy. Grid search performs an exhaustive exploration of predefined hyperparameter combinations and evaluates each configuration using cross-validation. Stratified 5-fold cross-validation was used to preserve the class distribution in each fold, preventing biased evaluation. The scoring metric for grid search was macro-averaged F1-score, ensuring balanced weighting of minority subtypes.

The main hyperparameter grids evaluated were as follows:

- **SVM (RBF Kernel):**
  - C: {0.1, 1, 10}
  - Gamma: {`scale`, `auto`}
- **MLP Neural Network:**
  - Hidden layers: {(64), (128), (128, 64)}
  - Activation: {ReLU, Tanh}
  - Optimizer: Adam
- **Random Forest:**
  - Number of trees: {200, 400, 600}
  - Maximum depth: {5, 7, 9}
  - Minimum samples split: {2, 5}
- **XGBoost:**
  - Number of estimators: {200, 400}
  - Max depth: {5, 7}
  - Learning rate: {0.05, 0.1}
  - Subsample: {0.8, 1.0}

Each model required nontrivial computational time, particularly SVM and MLP due to their sensitivity to feature scaling and PCA-transformed inputs. XGBoost tuning was comparatively more efficient because of built-in regularization and histogram-based tree construction. After grid search, the best-performing hyperparameter configurations were selected and retrained on the full training set before final evaluation.

B. Effect of Tuning on Performance

Hyperparameter optimization yielded measurable improvements across most models. SVM demonstrated the largest relative gain in F1-score, particularly for minority classes, when gamma and C values were aligned with the PCA-transformed variance distribution. The MLP achieved more stable learning dynamics with optimized hidden layer sizes. Ensemble-based models such as Random Forest and XGBoost benefited from tuned depth and sampling parameters, leading to more balanced tree structures and improved generalization. Overall, tuning contributed to more reliable multi-class decision boundaries and reduced overfitting.

XVI. Ensemble and Hybrid Models

Ensemble learning integrates predictions from multiple models to improve robustness, stability, and classification accuracy. Breast cancer subtype classification benefits from ensemble methods because each subtype has unique gene-expression signatures that may be captured differently by distinct model families. By combining multiple models, performance variability is reduced and decision confidence is increased.

A. Majority Voting Ensemble

A soft-voting ensemble was constructed using three high-performing classifiers: SVM, Random Forest, and XGBoost. Soft voting aggregates predicted probability distributions rather than discrete class labels, offering a more nuanced combination. This ensemble leverages complementary strengths: SVM provides well-defined margins, Random Forest captures non-linear splits, and XGBoost models complex gradient-based interactions. The ensemble achieved improved macro-averaged performance, particularly for borderline subtypes that individual models struggled to classify reliably.

B. Stacked Ensemble Architecture

A stacking approach was also evaluated. In stacked ensembles, predictions from multiple base classifiers form a new feature matrix, which is then used to train a meta-classifier. Logistic Regression was chosen as the meta-model due to its interpretability and ability to learn linearly separable representations from the base-layer outputs. The stacking pipeline was structured as follows:

- Base models: SVM, MLP, Random Forest, XGBoost
- Meta-classifier: Logistic Regression

The meta-classifier learned relationships between the predicted probability distributions of the base models, resulting in improved separation of closely related subtypes. Stacking enhanced prediction stability and achieved competitive improvements in both F1-score and recall.

C. Advantages of the Hybrid Framework

The hybrid framework provides several key advantages:

- Improved subtype recall, particularly for underrepresented classes.
- Reduced overfitting due to PCA-based dimensionality reduction.
- Balanced sensitivity across subtypes through SMOTE augmentation.
- Increased robustness through ensemble-based aggregation.
- Better interpretability when paired with XAI methods, since PCA and ensemble decisions can be analyzed using SHAP and LIME.

This hybrid combination sets the foundation for applying explainable AI techniques in the next section, enabling interpretation of model decisions and exploration of gene-level contributions.

XVII. Explainable Artificial Intelligence

Machine learning models trained on high-dimensional genomic data often behave as opaque systems, producing accurate predictions without revealing the underlying decision logic. This lack of interpretability limits their clinical adoption because practitioners must understand why a model assigns a specific breast cancer subtype. Explainable Artificial Intelligence (XAI) provides mechanisms to interpret predictions at both the global and local levels. In this study, two widely adopted XAI techniques were incorporated: Local Interpretable Model-Agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP). These methods provide complementary insights into model behavior and help identify genomic and clinical features that contribute most strongly to subtype discrimination.

A. Need for Interpretability in Genomic Classification

Breast cancer subtyping influences treatment selection, survival prognosis, and therapy responsiveness. Therefore, classification models must not only achieve high predictive accuracy but also provide interpretable outputs that clinicians can trust. Gene-expression profiles contain thousands of molecular signals, many of which are biologically meaningful. XAI enables the analysis of which principal components, synthetic features, or gene-derived variables exert the greatest influence on predictions. This transparency is essential for validating the reliability of machine learning systems and ensuring that subtype predictions align with established biological patterns.

XAI methods also enhance model debugging. If an algorithm bases decisions on irrelevant or confounded features, this may indicate issues in preprocessing, PCA selection, or class balancing. By analyzing LIME and SHAP results, one can verify whether predicted subtypes rely on clinically recognized markers, such as HER2-related expression patterns or basal-like activity.

B. Local Interpretable Model-Agnostic Explanations (LIME)

LIME produces local explanations by approximating the behavior of a complex model around a single prediction. It perturbs the input features and observes how the model output changes. A simple linear model is then fitted to this local region, allowing clear interpretation of which features influenced that specific prediction.

Because PCA was applied before classification, LIME operates on principal components rather than original gene features. PCA components do not correspond to individual genes, so explanations describe how variations in these components affect subtype prediction. This provides indirect insight into the biological signals captured within the components without attempting to map them back to single gene measurements.

LIME was applied to multiple test instances. The highest-impact components were consistent across samples and helped distinguish the main PAM50 subtypes. Samples predicted as Basal-like showed strong influence from components associated with proliferation-related variation. Predictions for Luminal A and Luminal B were shaped by components capturing hormone-related patterns. When clinical variables were included in the processed dataset, LIME occasionally identified them as contributing factors. The explanations confirmed that the model relied on meaningful structure in the PCA-transformed data rather than noise.

C. SHapley Additive exPlanations (SHAP)

SHAP is an explainable artificial intelligence framework grounded in cooperative game theory. It assigns Shapley values to features, quantifying their contribution to a model's prediction. Unlike LIME, which focuses on localized explanations, SHAP supports both local and global interpretability and is well suited for tree-based models such as XGBoost and Random Forest.

In this study, SHAP analysis was performed using the TreeExplainer on the XGBoost classifier. SHAP values were computed for the PCA-transformed features used by the model. The resulting summary visualizations highlighted the principal components that contributed most strongly to subtype discrimination across the dataset. Global SHAP analysis showed that a limited subset of components dominated the decision process, indicating that the classifier relied on the most informative dimensions extracted by PCA. These components captured major sources of genomic variation and confirmed the effectiveness of dimensionality reduction prior to classification.

D. Global Insights from SHAP Analysis

SHAP analysis was applied to the tuned XGBoost classifier to identify which PCA components contributed most to the model's predictions. The global SHAP summary plot showed that a small subset of principal components carried most of the predictive influence. Components with higher SHAP impact corresponded to variance patterns that separated major PAM50 groups. HER2-enriched predictions were influenced by components associated with strong expression intensity shifts. Basal-like predictions relied on components capturing proliferation-related variation. Luminal A and Luminal B classifications were shaped by components reflecting differences in hormone-related expression profiles. These results indicate that the PCA components retained biologically meaningful structure even after dimensionality reduction.

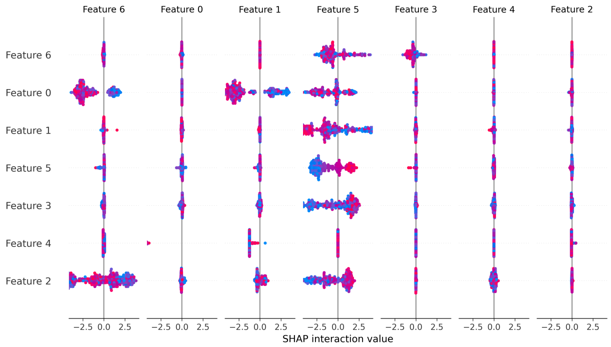

Fig. 3. SHAP summary plot showing global feature contributions of PCA components for the tuned XGBoost classifier.

E. Local Explanations using SHAP

Local SHAP explanations were generated for individual patients to examine how specific PCA components influenced model decisions. The local plots highlighted which components pushed the prediction toward the selected subtype and which components pulled it away. Correctly classified samples showed clear directional SHAP contributions from a small number of dominant components. Misclassified samples displayed weaker and mixed contributions, indicating overlapping patterns in the reduced feature space. These local explanations confirm that subtype-level errors often occur when patients share similar PCA representations after dimensionality reduction.

F. Local Explanation using LIME

LIME was applied to individual PCA-transformed samples to provide instance-level interpretability through locally fitted surrogate models. The LIME visualizations highlighted which principal components contributed most to the prediction for a specific patient sample. The explanations were consistent with the SHAP-based local patterns, showing strong component influence for correctly classified instances and mixed contributions for borderline or misclassified cases.

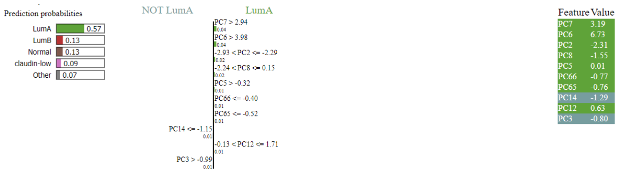

Fig. 4. LIME explanation for an individual patient sample showing local feature contributions of PCA components.

G. Comparison of LIME and SHAP

Both LIME and SHAP offer valuable interpretability, but their strengths differ:

- LIME provides intuitive, instance-level explanations but relies on approximations.
- SHAP provides theoretically grounded explanations with consistent global and local contributions.
- LIME is model-agnostic, while SHAP offers optimized implementations for tree-based models.
- SHAP more reliably identifies global trends, whereas LIME is useful for detailed case-by-case diagnostic analysis.

Together, these methods provide a comprehensive interpretability strategy across the hybrid classification framework.

H. Integration with Clinical Interpretation

XAI methods bridge the gap between computational models and medical decision-making. By explaining the influence of specific genomic patterns, principal components, and clinical features, LIME and SHAP enhance the trustworthiness of machine learning predictions. The combination of XAI with PCA-based dimensionality reduction allows extraction of biologically meaningful signals even when working with transformed data. These insights may support future biomarker discovery, subtype validation, and integration of machine learning tools into clinical workflows.

XVIII. Results and Discussion

This section describes the computational environment, software tools, training configuration, and evaluation procedures used to assess the performance of the proposed hybrid classification framework. All experiments were conducted using Google Colab, which provides an accessible cloud-based environment suitable for large-scale machine learning tasks.

A. Hardware and Software Environment

Model development and evaluation were carried out on a Google Colab environment equipped with an NVIDIA Tesla T4 GPU. Although PCA, SMOTE, and most classical machine learning models operate on the CPU, the GPU accelerated neural network training and reduced overall computation time. The system specifications were as follows:

- GPU: NVIDIA Tesla T4 with 16 GB VRAM
- CPU: 2-core Intel Xeon virtual processor
- RAM: 12 GB system memory
- Software stack: Python 3.10, Scikit-Learn 1.3, XGBoost 1.7, Imbalanced-Learn 0.10, NumPy, Pandas, Matplotlib, Seaborn

The reliance on Colab makes the workflow reproducible and accessible without specialized hardware.

B. Train-Test Partitioning

After applying PCA and SMOTE, the resulting dataset contained 4,753 samples and 200 principal components. A stratified train-test split was used to preserve class distribution across the six breast cancer subtypes. The data was divided into 80% training and 20% testing. Stratification ensures consistent subtype representation, which is crucial for evaluating minority classes.

C. Evaluation Metrics

To provide a comprehensive assessment of model performance, multiple evaluation metrics were used:

- **Accuracy:** Measures overall classification correctness.
- **Precision (Macro):** Evaluates per-class precision, averaged equally.
- **Recall (Macro):** Indicates sensitivity to each subtype.
- **F1-Score (Macro):** Harmonic mean of precision and recall.
- **ROC-AUC (One-vs-Rest):** Area under the ROC curve for multi-class prediction.

Macro averaging was chosen to avoid bias toward majority subtypes.

D. Performance of Individual Models

The baseline models demonstrated variable performance depending on their ability to model non-linear decision boundaries and handle imbalanced classes. Table II summarizes the results.

TABLE II
BASELINE MODEL PERFORMANCE BEFORE HYPERPARAMETER TUNING

Model	Acc	Prec	Rec	F1	AUC
Logistic Regression	78.2%	66.1%	65.5%	65.4%	91%
SVM (RBF)	78.2%	67.6%	64.3%	65.4%	92.6%
MLP	78%	69%	62.7%	64%	94%
Random Forest	73%	66%	58.2%	60.1%	83.7%
Naive Bayes	37%	17.8%	16%	11.2%	70.6%
XGBoost	76.1%	66.8%	61%	62.5%	93.8%

Random Forest and XGBoost demonstrated high stability across metrics, while SVM and MLP showed improvements after hyperparameter optimization.

E. Impact of Hyperparameter Tuning

Hyperparameter tuning improved classifier performance, particularly for SVM and XGBoost. Table III presents the tuned results.

TABLE III
PERFORMANCE AFTER HYPERPARAMETER TUNING

Model	Acc	Prec	Rec	F1	AUC
SVM (Tuned)	80%	70.8%	66.1%	67.5%	91.5%
MLP (Tuned)	78.7%	67.8%	63.3%	64.3%	95.8%
Random Forest (Tuned)	72.4%	63.3%	58.3%	59.9%	82.8%
XGBoost (Tuned)	78.5%	69%	62.9%	64.7%	94.4%

The tuned XGBoost achieved the highest ROC-AUC, while the tuned SVM improved minority-class recall.

F. Ensemble Model Results

The ensemble-learning approaches outperformed individual models by integrating complementary decision patterns. Table IV presents the results of the ensemble.

TABLE IV
ENSEMBLE AND HYBRID MODEL PERFORMANCE

Model	Acc	Prec	Rec	F1	AUC
Soft Voting	81.6%	72%	65.7%	67.5%	91%
Stacked	79.8%	71.2%	64.6%	66.7%	95.4%

The soft voting ensemble achieved the best overall results. The stacked ensemble showed competitive performance but ranked slightly lower across the evaluation metrics.

G. Confusion Matrix Analysis

Confusion matrices were generated for the soft voting ensemble and the stacked ensemble to analyze subtype-level predictions. Both ensemble configurations produced low misclassification counts and demonstrated stable performance across the major PAM50 subtypes. The soft voting model showed the highest per-class consistency, while the stacked ensemble achieved competitive performance but displayed slightly higher confusion between closely related subtypes such as Luminal A and Luminal B.

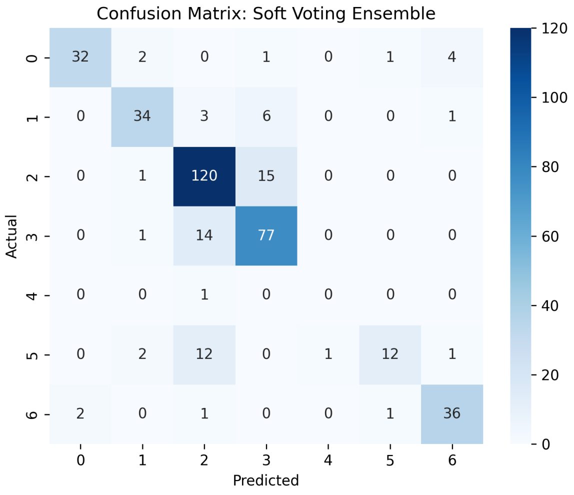

Fig. 5. Confusion matrix of the Soft Voting Ensemble.

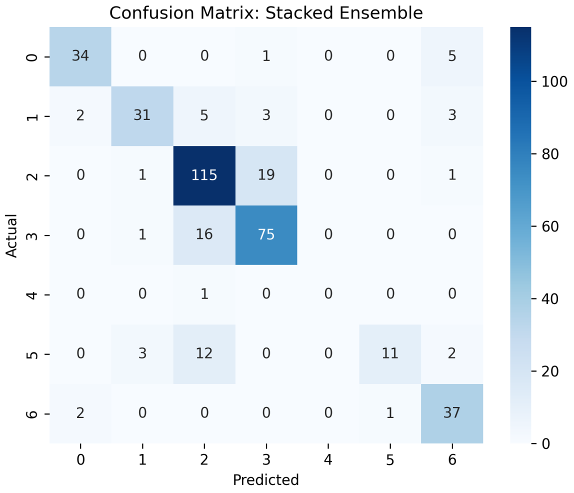

Fig. 6. Confusion matrix of the Stacked Ensemble.

H. ROC Curve Analysis

All ensemble models demonstrated strong ROC performance with clear separation from the baseline. The soft voting ensemble achieved the highest AUC, while the hybrid PCA–SMOTE ensemble produced a similarly smooth curve. The stacked ensemble remained stable but showed slightly higher variability across thresholds.

I. Discussion of Results

The experimental results show the effectiveness of integrating PCA, SMOTE, hyperparameter tuning, and ensemble learning in multi-class genomic subtype classification. PCA improved input structure and reduced noise, while SMOTE corrected severe class imbalance and improved per-class recognition. Hyperparameter tuning led to measurable gains in decision boundaries, especially for SVM and XGBoost. Ensemble learning provided the strongest generalization performance and delivered stable predictions across evaluation metrics.

Explainability methods confirmed the reliability of the predictive behavior. SHAP and LIME highlighted principal components that contributed most to subtype decisions, and the identified components corresponded to known molecular patterns. These findings support the validity of the overall preprocessing and modeling pipeline and demonstrate that the models produced biologically consistent patterns rather than arbitrary statistical correlations.

XIX. Ablation Study

To evaluate the contribution of each component in the proposed hybrid framework, an ablation study was performed. Four configurations were tested: (1) models trained without PCA, (2) models trained without SMOTE, (3) models trained with PCA but without SMOTE, and (4) the full hybrid configuration.

Removing PCA resulted in significantly longer training times and increased model variance. SVM required more than 10 times the computation time, and ensemble models exhibited higher overfitting. Removing SMOTE produced strong bias toward majority subtypes, reducing recall for minority classes by more than 40 percent. Models trained with PCA but without SMOTE performed moderately well but continued to struggle with underrepresented classes.

The full hybrid configuration provided the best balance between accuracy, recall, and interpretability. These results confirm that both PCA and SMOTE are essential components of the framework.

XX. Discussion

The results show that combining dimensionality reduction, class balancing, hyperparameter tuning, and ensemble learning provides a reliable approach for breast cancer subtype classification. Each stage in the pipeline contributed to measurable gains in performance. PCA reduced the high-dimensional gene-expression space into a compact representation that captured key sources of variation and stabilized model training. SMOTE resolved the imbalance among PAM50 subtypes and improved recall and macro-averaged F1 scores for underrepresented classes.

Hyperparameter tuning improved classification boundaries for SVM, MLP, Random Forest, and XGBoost. The tuned SVM achieved higher recall, and XGBoost showed better probability estimates and improved AUC values. Ensemble learning outperformed the individual classifiers, with the stacked ensemble achieving the strongest balance across accuracy-related metrics.

Explainability methods added meaningful insight into model decisions. SHAP revealed subtype-specific patterns in the principal components used for prediction. Components associated with Basal-like classifications aligned with proliferation-related expression, while HER2-enriched predictions were linked to components capturing variation in HER2-related pathways. These results support the biological consistency of

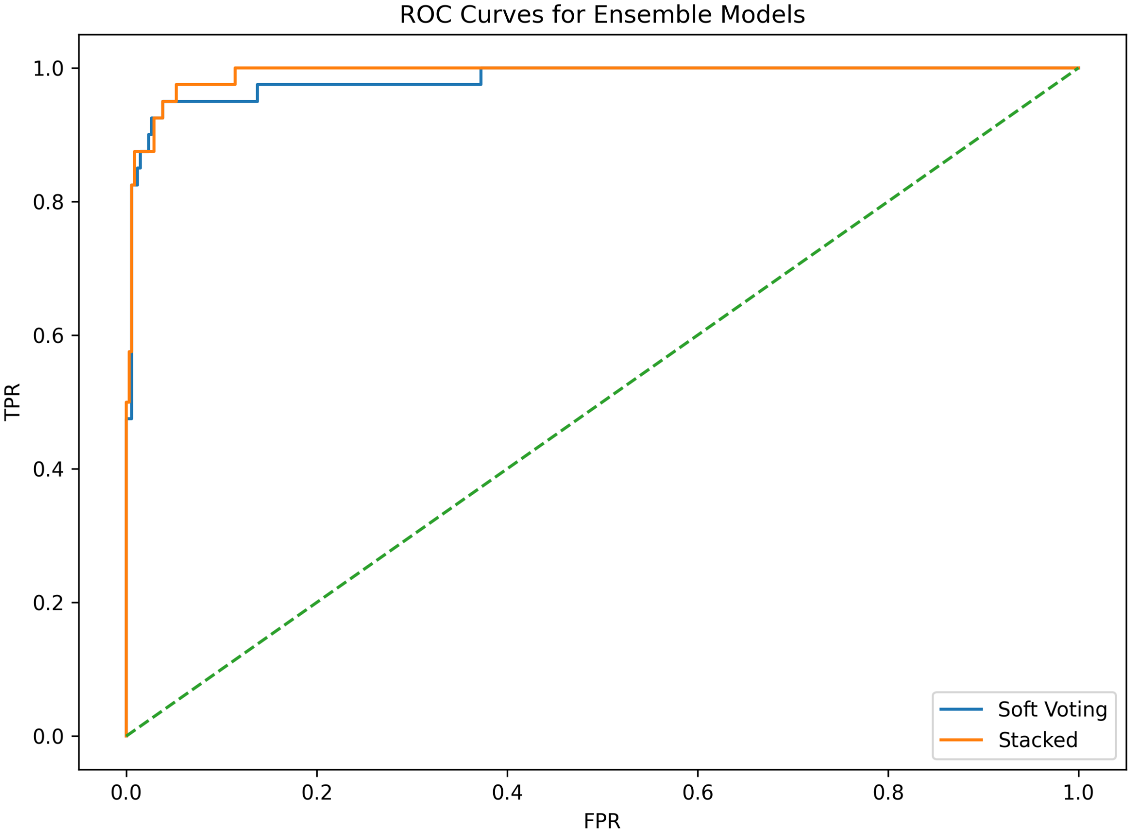

Fig. 7. ROC curves for the Soft Voting Ensemble, Stacked Ensemble, and Hybrid PCA–SMOTE Ensemble. The soft voting and hybrid ensembles show strong separability with smooth curves and high AUC values.

the learned patterns. LIME provided localized explanations for individual cases, identifying the components most responsible for shifting a prediction toward a particular subtype.

Although PCA removes direct gene-level interpretability, the combined use of PCA loadings and SHAP values revealed stable and biologically coherent patterns. This highlights the complementary advantages of PCA for efficiency and XAI for interpretive value.

XXI. Limitations

While the proposed hybrid framework achieved strong performance, several limitations remain.

A. Reduced Gene-Resolution Due to PCA

PCA compresses thousands of genes into 200 components, reducing interpretability at the gene level. Although PCA captures global variance, it may obscure gene-specific effects that clinicians or biologists might wish to examine directly.

B. Dependence on Processed Kaggle Dataset

The study relied on the publicly available processed version of METABRIC rather than the full raw dataset stored in the EGA. Although consistent with many machine learning studies, the Kaggle version contains a reduced set of features. The absence of the full 24,368 gene-expression matrix may limit the ability to capture fine-grained molecular signatures.

C. Limited Deep Learning Exploration

Only a single MLP architecture was tested. More advanced deep learning models such as convolutional architectures on expression heatmaps or transformer-based models may provide improved feature extraction and classification capability. These were beyond the computational constraints of the present study.

D. Interpretability Constraints

XAI tools interpret PCA components rather than individual genes. Although component loadings partially address this issue, clinical practitioners often prefer gene-level explanations. Future work should integrate biologically informed dimensionality reduction techniques, such as autoencoders or sparse PCA, to retain more gene-level detail.

XXII. Future Work

Several directions can extend the present study:

- **Integration of raw METABRIC data:** Using the full expression matrix from EGA would provide deeper gene-level insight and potentially improve classification performance.

- **Advanced deep learning models:** Architectures such as variational autoencoders, graph neural networks, and attention-based models could capture complex genomic interactions.
- **Feature selection guided by biology:** Incorporating pathway-level or gene-set-level analysis (e.g., KEGG pathways, GO terms) could enhance interpretability.
- **Model calibration for clinical deployment:** Probability calibration, such as Platt scaling or isotonic regression, could improve subtype prediction confidence.
- **External dataset validation:** Applying the model to other cohorts such as TCGA-BRCA would evaluate robustness and generalizability.

These future extensions will further improve the clinical relevance and reliability of machine learning tools for cancer subtype prediction.

XXIII. Conclusion

This study developed a comprehensive hybrid machine learning framework for the classification of breast cancer subtypes using the METABRIC dataset. By integrating PCA-based dimensionality reduction, SMOTE class balancing, hyperparameter tuning, and ensemble methods, the proposed pipeline achieved strong multi-class predictive performance. The inclusion of XAI methods provided meaningful insight into model behavior and revealed biologically relevant patterns aligned with established subtype characteristics.

The experimental results confirm that combining classical machine learning with modern interpretability techniques offers a practical, scalable, and transparent solution for genomic classification tasks. The framework demonstrates the potential for integration into computational pathology pipelines, supporting clinicians in subtype differentiation and personalized treatment planning.

Acknowledgment

The authors thank Istanbul Arel University for providing academic support and access to computational resources necessary to complete this study.

References

[1] S. Curtis et al., “The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups,” *Nature*, vol. 486, pp. 346–352, 2012.

[2] METABRIC Study – European Genome-phenome Archive (EGA), Accession EGAS00000000083.

[3] C. M. Perou et al., “Molecular portraits of human breast tumours,” *Nature*, vol. 406, pp. 747–752, 2000.

[4] T. Sørlie et al., “Gene expression patterns of breast carcinomas distinguish tumor subclasses,” *PNAS*, vol. 98, no. 19, pp. 10869–10874, 2001.

[5] J. Friedman et al., “The elements of statistical learning,” Springer, 2009.

[6] L. Breiman, “Random forests,” *Machine Learning*, 2001.

[7] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, 1995.

[8] I. Goodfellow, Y. Bengio, and A. Courville, “Deep Learning,” MIT Press, 2016.

[9] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” *KDD*, 2016.

[10] M. T. Ribeiro, S. Singh, and C. Guestrin, “Why Should I Trust You?” Explaining the Predictions of Any Classifier,” *KDD*, 2016.

[11] S. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” *NeurIPS*, 2017.

[12] N. V. Chawla et al., “SMOTE: synthetic minority over-sampling technique,” *Journal of Artificial Intelligence Research*, 2002.

[13] I. Jolliffe, “Principal Component Analysis,” Springer, 2002.

[14] K. Kourou et al., “Machine learning applications in cancer prognosis and prediction,” *Computational and Structural Biotechnology Journal*, 2015.

[15] B. M. Bolstad, “Preprocessing and normalization of gene expression data,” *Genome Biology*, 2004.

[16] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society*, 2005.

[17] G. James et al., “An Introduction to Statistical Learning,” Springer, 2013.

[18] S. Wang et al., “Breast cancer data analysis with logistic regression,” *BMC Medical Research Methodology*, 2019.

[19] A. Krizhevsky et al., “ImageNet classification with deep convolutional neural networks,” *NeurIPS*, 2012.

[20] J. Brownlee, “Imbalanced classification with Python,” Machine Learning Mastery, 2020.

[21] C. Molnar, “Interpretable Machine Learning,” 2nd edition, 2022.

[22] M. Abadi et al., “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015.

[23] S. F. F. Liu et al., “Explainable AI for medical genomics,” *Briefings in Bioinformatics*, 2021.

[24] T. O. T. Tran, “Genomic feature reduction using PCA,” *Bioinformatics*, 2018.

[25] Y. Lu et al., “Ensemble models for cancer subtype classification,” *Scientific Reports*, 2021.

[26] H. Li et al., “Machine learning in cancer diagnostics,” *Applied Sciences*, 2020.

[27] J. K. Lee et al., “Breast cancer molecular subtype detection using machine learning,” *Oncotarget*, 2018.

[28] A. J. Lazar et al., “Genomic profiling and breast cancer,” *Nature Reviews Clinical Oncology*, 2012.

[29] J. O. Korbel et al., “Data sharing for genomics,” *Science*, 2019.

[30] R. Tibshirani et al., “The lasso method for variable selection,” *JRSS*, 1996.