# Beyond PAM50: Novel Breast Cancer Subtypes from BIGCLAM-Based Graph Clustering

**Saeed Samimi [1], Saeed Pirmoradi [2], Mohammad Teshnehlab [1*]**

[1]Department of Electric and Computer Engineering, K.N. Toosi University of Technology, Tehran, Iran.

[2]Clinical Research Development Unit of Tabriz Valiasr Hospital, Tabriz University of Medical Sciences, Tabriz, Iran.

* Corresponding author at [1]Department of Electric and Computer Engineering, K.N. Toosi University of Technology, Tehran, Iran.
E-mail address: teshnehlab@eetd.kntu.ac.ir (M. Teshnehlab)
Other authors' e-mail addresses: s.samimi@email.kntu.ac.ir (S. Samimi), SAID.PIRMORADI@gmail.com (S. Pirmoradi)

**Abstract:**

Breast cancer is a complex and heterogeneous disease with diverse molecular profiles. Traditional subtyping systems, such as PAM50, have advanced diagnosis and treatment selection but may not fully capture the intricate structure of gene expression data. In this study, we propose a novel, unsupervised methodology to discover new breast cancer subtypes using gene expression data and the BIGraph Cluster Affiliation Model (BIGCLAM), originally developed for overlapping community detection in networks. Rather than assuming predefined subtype labels, our approach identifies intrinsic clusters within high-dimensional gene expression profiles. These discovered subtypes differ from conventional classifications and exhibit stronger internal consistency. We evaluated their discriminative power using downstream classification with multi-layer perceptron (MLP) and support vector machine (SVM) models. Our method achieved higher accuracy than two widely used PAM50-based classifiers—AIMS and miniABS—when applied to the same dataset (GSE96058). These results suggest that BIGCLAM-derived subtypes may better reflect underlying biological variation, offering new insights for personalized medicine and breast cancer treatment stratification.

**Keywords:** Breast cancer Subtypes; Overlapping community Detection; Graph clustering; Machine Learning; Artificial Neural Network

## 1. Introduction

One of the deadliest forms of cancer is breast cancer, which is defined by the unchecked proliferation of cells in the breast tissue. With an estimated 42,170 people (42,170 women and 520 males) projected to die from breast cancer in the United States in 2019 and roughly 268,600 new cases (268,600 women and 2,670 men) anticipated to occur, breast cancer continues to be a leading cause of cancer-related fatalities [1]. The difficulty in identifying the disease in its early stages with conventional clinical methods is one of the major obstacles in managing breast cancer[2]. As a result, many breast cancer patients have few post-surgery treatment options, which frequently leads to a survival duration of less than a year. Consequently, the fundamental goal of early-stage breast cancer detection is to improve patient outcomes by enabling timely and effective treatment options.

Finding biomarkers to aid in early detection and treatment of breast cancer has been a focus of recent research. reduce side effects and improve survival rates, medical professionals are developing new methods for precise diagnosis and identifying its subtypes [3]. Genetic data is being used more and more to distinguish between various breast cancer subtypes in addition to clinical information. Gene expression is a key

indicator; it has sparked research into the identification and detection of different malignancies, especially with the improvements of the last decade in the extraction of gene expression information [4].

Small non-coding RNAs with 19–25 nucleotides are known as miRNAs. They are essential for basic biological functions as metabolism, apoptosis, proliferation, and cell cycle regulation, playing critical roles in maintaining cellular homeostasis and function. [5]. Thousands of miRNAs control over 60% of the protein-coding genes in the human genome [6], [7]. Although numerous studies have linked miRNA dysregulation to various cancer types, much remains unknown about their properties, including their specific roles in different cancer subtypes and their potential as therapeutic targets. [8].

In our study, we focus on gene expression profiles instead of miRNA data. The process of using a gene's instructions to create a functional gene product—usually a protein—is known as gene expression. Because they provide important insights into the molecular mechanisms driving carcinogenesis, gene expression profiles have been used extensively in cancer research. As an example, previous researchers show that gene expression patterns can be used to identify different subtypes of breast cancer, with varying consequences for prognosis [9]. Subsequent research has validated the usefulness of gene expression in improving the categorization of breast cancer and directing therapeutic choices [5,10].

The PAM50 method is a well-known technique among the numerous approaches established for breast cancer subtyping. Based on the expression of 50 genes, the PAM50 assay divides breast cancer into intrinsic subtypes: Luminal A, Luminal B, HER2-enriched, Basal-like, and Normal-like. This method has proven to be a robust tool for both prognostication and therapeutic decision-making, significantly improving the accuracy of breast cancer subtype identification [10]. Recent research has continued to build on the PAM50 framework, integrating new genomic data and refining classification algorithms to enhance their clinical relevance. For example, previous works have discussed the evolving landscape of breast cancer treatment, emphasizing the role of molecular subtyping in personalizing therapy.[11].

In addition to PAM50, our study also considers recent advancements in breast cancer subtyping. Methods like AIMS (Absolute Intrinsic Molecular Subtyping) and miniABS have been developed to address limitations in existing classification systems. AIMS utilizes a refined set of molecular features to offer more precise subtype definitions, while miniABS provides a computationally efficient alternative without sacrificing accuracy [12,13]. Both methods have demonstrated significant promise in recent studies, with applications ranging from clinical diagnostics to treatment planning.

Despite these developments, PAM50 and its derivatives rely on static gene signatures and predefined subtype boundaries. This may obscure alternative biological stratifications present in gene expression data. In this study, we adopt a different strategy: instead of replicating PAM50 labels, we use BIGCLAM to discover novel molecular subgroups via graph-based clustering. We then assess the predictive strength of these new subtypes and compare their performance against PAM50-defined labels in downstream classification.

To determine which gene expressions are the most discriminant for each subtype, we employ a feature selection method that leverages variance measurements. Using a classifier, we construct a graph based on these selected gene expressions. After employing the graph and BIGCLAM approach for breast cancer classification, we applied classifier methods. The classifier accurately distinguishes breast cancer subtypes when the appropriate gene expressions are selected. We propose using Multi-Layer Perceptron (MLP) and multi-layer Support Vector Machine (SVM) as the classifiers. The combination of these techniques is well-suited for genomic data with high dimensionality.

The remainder of this essay is structured as follows: Details of the dataset are provided in **Section 2**. The methodology applied to the breast cancer gene expression data is described in **Section 3**. In this section, **Section 3.1** covers the feature selection process, **Section 3.2** explains the graph representation of the data, **Section 3.3** discusses the clustering algorithm, **Section 3.4** addresses the data augmentation techniques used for class imbalance and **Section 3.5** focuses on the classifiers employed in the study. **Section 4** presents the experimental results and analysis, followed by a conclusion in **Section 5**.

## 2. Data

The GSE96058 dataset, which is publicly accessible via the Gene Expression Omnibus (GEO) repository[14], was used for this investigation. This dataset, which includes RNA sequencing data from patients with breast cancer, was thoroughly analyzed and documented by reserachers[15]. Comprehensive RNA sequencing data from a population-based multicenter cohort participating in the Sweden Cancerome Analysis Network—Breast Initiative (SCAN-B) is part of this dataset.

The main aim of the SCAN-B study was to assess the clinical utility of RNA sequencing-based classifiers to predict five standard biomarkers for breast cancer: progesterone receptor (PR), HER2, estrogen receptor (ER), Ki-67, and Nottingham histologic grade (NHG). The cohort provides a robust foundation for our study, as it includes samples from a diverse range of patient populations[15].

## 3. Methodology

Figure 1 illustrates the entire workflow, which consists of six distinct stages. In the first stage, a suitable feature selection algorithm is employed to identify candidate gene expressions with significant discriminative power. Next, the data is transformed into a graph representation. The BIGCLAM algorithm is then applied to cluster the graph-structured data. In the fourth stage, due to the imbalance in the data, augmentation techniques are used to balance the dataset. Finally, classifiers categorize breast cancer subtypes based on the selected gene expressions and the clusters derived from the preceding steps.


**Figure 1:** The whole process block diagram

### 3.1. Feature Selection

Many redundant and irrelevant genes that do not impact the disease can make high-dimensional data, such as gene expression data with around 30,865 features, less effective in classifying diseases or subtypes. As a result, feature selection (FS) methods are essential for removing duplicate and unnecessary features. Three categories are used to categorize FS algorithms: filter, wrapper, and embedding techniques.

Filter methods identify features individually and assign weights based on their importance. These techniques are classifier-independent and do not require a classifier. Wrapper methods are dependent on the classifier since they take feature interactions into account and use a classifier to evaluate subsets of features. Embedded methods perform FS during training and consider feature interactions.

Due to the low computational cost of filter methods, especially when dealing with high-dimensional data, they were used in this study to perform FS. Many filtering methods have been proposed recently, and they

can be divided into three groups based on the relevance metric: information theory, correlation, and distances between distributions [16]. In this paper, the variance-based filter strategy was used in the selection of features since calculating entropy and probability density functions can be skewed in high-dimensional sparse data. Therefore, for FS approaches based on distribution distances and information theory perform better with dense data.

The specific method used for feature selection in this study is the Variance Threshold method. This method removes all features whose variance does not meet a certain threshold, thus eliminating features with low variability. utilizing a process of try and error, we determined that a threshold value of 13 was optimal for our dataset. The variance of a feature $i$ is calculated as follows:

$$Variance_i = \frac{1}{N}\sum_{j=1}^{N}(x_{ij} - \bar{x}_i)^2 \quad (1)$$

Where $x_{ij}$ is the value of feature $i$ for sample $j$, $\bar{x}_i$ is the mean value of feature $i$, $N$ is the total number of samples. Features with a variance below the threshold value of 13 were removed from the dataset. This step ensured that only features with substantial variability across samples were retained, improving the efficiency and performance of subsequent analyses

### 3.2. Graph Representation

After performing feature selection, the data was represented as a graph to facilitate the use of the BigClam algorithm in the next section. The process began with loading the dataset into a DataFrame. To determine relationships between data points, cosine similarity was computed between feature vectors, excluding the community label column. Cosine similarity is a widely used metric for measuring the similarity between two non-zero vectors, defined as follows:

$$Cosine\_Similarity(A, B) = \frac{A \cdot B}{||A||||B||} \quad (2)$$

where $A$ and $B$ are the feature vectors of two data points, $A \cdot B$ is the dot product of $A$ and $B$, and $||A||$ and $||B||$ are the magnitudes (Euclidean norms) of $A$ and $B$, respectively [17].

An adjacency matrix was created by applying a threshold to the cosine similarity score. A threshold of 0.5 was chosen based on empirical analysis and domain knowledge in this study. Specifically, this threshold was determined by analyzing the distribution of cosine similarity scores across the dataset and selecting a value that balances the inclusion of meaningful connections while excluding weaker, less significant ones. Edges were formed between data points if their similarity exceeded this threshold. Self-loops were removed by subtracting the identity matrix from the adjacency matrix. Community assignments were extracted from the last column of the DataFrame, associating each data point with a set representing its community.

The graph was represented as a JSON object, with each node labeled according to its community information. Links, or edges, were created between nodes that had a similarity score above the threshold. Each edge was assigned a value and distance based on the intersection of community sets, with these attributes scaled to improve visualization and analysis. The resulting graph data was saved in multiple formats for further use, including JSON for visualization purposes and binary formats for the adjacency matrix and community assignments.

This graph representation method provided a structured and efficient way to visualize and analyze data point relationships. Using cosine similarity retained only meaningful connections, offering a clear depiction

of the underlying community structure [18]. This approach supports the analysis of overlapping communities, which is essential for algorithms like BIGCLAM [19].

### 3.3. BIGCLAM Clustering

In this study, we employ the BIGCLAM algorithm to perform community detection on our graph. The BIGCLAM algorithm, developed by Yang and Leskovec (2013), is designed to detect overlapping communities in large-scale networks.

BIGCLAM models the community memberships of nodes through a membership matrix $F$. Each element $F_{ik}$ represents the strength of node $i$'s affiliation to community $k$. The probability of an edge existing between nodes $i$ and $j$ is modeled follows:

$$P(i,j) = 1 - exp(-\sum_{k} F_{ik}F_{jk}) \tag{3}$$

This formulation allows for overlapping communities, enabling nodes to have non-zero affiliations with multiple communities.

The likelihood of the observed graph given the membership matrix $F$ is defined as:

$$L(F) = \sum_{(i,j)\in E} log(1 - exp(-F_i \cdot F_j)) - \sum_{(i,j)\notin E} F_i \cdot F_j \tag{4}$$

Maximizing this likelihood function ensures the model accurately reflects the network structure. The first term rewards edges between nodes with strong community affiliations, while the second term penalizes the lack of such edges.

To maximize the log-likelihood function, we use gradient ascent. The gradient of the log-likelihood concerning $F_{ik}$ is given by:

$$\frac{\partial L}{\partial F_{ik}} = \sum_{j\in N(i)} \left(\frac{F_{jk}\, exp(-F_i \cdot F_j)}{1 - exp(-F_i \cdot F_j)}\right) - \sum_{j\notin N(i)} F_{jk} \tag{5}$$

Where $N(i)$ denotes the set of neighbors of node $i$. This gradient helps in updating the membership matrix $F$ iteratively.

The training process involves, initializing the membership matrix $F$ randomly, and updating it using the gradient ascent method. The steps are as follows:

1. Initialization: $F$ initialized with random values.

2. Iteration: For each node $i$ and each community $k$, $F_{ik}$ updated using the gradient of the log-likelihood gradiant.

3. Convergence: The process is repeated for a fixed number of iterations or until convergence.

The optimal number of communities is determined using the Akaike Information Criterion (AIC), which calculated as:

$$AIC = -2L(F) + 2K \tag{6}$$

Where $k$ is the number of parameters in the model, this criterion balances the model's goodness of fit with its complexity.

Table 1 displays the pseudo-code for estimating the optimal number of communities using BIGCLAM.

**Table 1:** Pseudo-code for estimating the ideal number of communities using BIGCLAM

**Input:**

Adjacency matrix A (size N x N), Max number of communities: max_communities, Number of iterations: iterations

**Output:**

Best membership matrix F, Best number of communities

**Initialization:**

1. Initialize F with random values (size N x num_communities).

**Iteration:**

**2. For k := 1 to max_communities do begin**
   a. Initialize F with random values (size N x k).
   b. For iter := 1 to iterations do begin
      i. For i := 1 to N do begin
         - Compute the gradient of the log-likelihood with respect to F_ik.
         - Update F_ik using the gradient ascent step.
         - Ensure F_ik remains nonnegative (F_ik = max(0.001, F_ik)).
      Endfor;
   Endfor;
   c. Compute the log-likelihood of the current F.
   d. Calculate the AIC for the current F.
   e. If current AIC < best_AIC then
      - best_AIC := current AIC.
      - best_F := current F.
      - best_num_communities := k.
   Endif;
Endfor;

**Return:**

best_F: The membership matrix with the best AIC.
best_num_communities: The number of communities corresponding to best_F.

After training the BIGCLAM model, the membership result, matrix $F$, analyzed to understand the community structure of the network. The affiliations of each node to various communities examined, and the model's performance is evaluated based on the log-likelihood and AIC values [20].

### 3.4. Data Augmentation

The data was supplemented before classification to equalize the number of occurrences in each class. The following steps were engaged in the augmentation process:

1.Determine how many instances there are in every class.

2. Establish the most instances that a class can have.

3. Adjust the other classes to correspond with this upper limit of occurrences.

Following that, the enlarged dataset was used for validation and training to guarantee a fair representation of all classes. In particular, an 80:10:10 ratio utilized to separate the data into test, validation, and training sets. The original (unaugmented) data was kept for testing and validation, whereas the augmented data mainly used for training and validation.

### 3.5. Classification

In this part, we outline the classification techniques used to assess the BIGCLAM algorithm's cluster' quality. For this, we specifically employed MLP and SVM. Our dataset was given new labels by the clusters produced by BIGCLAM, which allowed us to carry out supervised classification.

#### 3.5.1. SVM

A popular supervised learning technique for classification tasks is SVM. Finding the ideal hyperplane with the largest margin between data points of different classes is the aim of SVM. The data points closest to the decision boundary are called support vectors (SV) and are what define this hyperplane. SVM is a common option for classification applications due to its robustness against overfitting and performance in high-dimensional domains [21]. The optimization issue that the SVM method resolves is as follows:

$$min_{w,b,\xi} \frac{1}{2}||W||^2 + C\sum_{i=1}^{n} \xi_i \quad (7)$$

Subject to:

$$y_i(W \cdot X_i + b) \geq 1 - \xi_i \quad (8)$$

Where $W$ is the weight vector, $b$ is the bias term, $\xi_i$ are slack variables, $C$ is the regularization parameter, and $(X_i, y_i)$ are the training samples.

#### 3.5.2. MLP

An artificial neural network with many layers of neurons, comprising an input layer, one or more hidden layers, and an output layer, is called an MLP. Weighted connections bind every neuron in one layer to every other neuron in the layer below it. Nonlinear activation functions used by MLPs to simulate intricate data interactions. Using backpropagation, the network trained by minimizing the error between the expected and actual outputs by modifying the weights [22].

The output of an MLP with one hidden layer can be express as:

$$y = f(W_2 \cdot g(W_1 \cdot X + b_1) + b_2) \quad (9)$$

Where$W_1$ and $W_2$ are the weight matrices for the hidden and output layers, $b_1$ and $b_2$ are the bias vectors, $g$ is the activation function for the hidden layer, and $f$ is the activation function for the output layer.

In this Study We Used 5 layer MLP with LeakyRelu Activation Function for Hidden Layers and Softmax Activation Function for Output Layer, Optimization Function was Adam and Learning Rate was 0.0005, number of Epochs was 1000 and we iterate the whole prosses for 100 times.

The Classification Procedure is shown in Table 2.

**Table 2:** Classification Procedure

**Input:**
Labeled dataset with clusters from BIGCLAM, graph with node attributes and structure

**Output:**
Performance metrics (accuracy, precision, recall, F1-score) for SVM and MLP

**Steps:**

**1. Data Preparation:**
1.1. For each node in the graph, assign the cluster label obtained from BIGCLAM as the new label for the node

**2. Feature Extraction:**
2.1. For each node in the graph, extract the feature vector based on node attributes and structural properties

**3. Training and Testing Split:**
3.1. Split the dataset into three parts: 80% for training, 10% for validation, and 10% for testing.

**4. Training the SVM:**
4.1. Initialize SVM with chosen hyperparameters
4.2. Train SVM using the training set
4.3. Perform cross-validation to determine optimal hyperparameters (e.g., C)

**5. Training the MLP:**
5.1. Initialize MLP with three hidden layers and chosen architecture
5.2. Train MLP using backpropagation and gradient descent optimization on the training set

**6. Evaluation:**
6.1. Evaluate SVM on the testing set and calculate accuracy, precision and recall, F1-score for SVM
6.2. Evaluate MLP on the testing set and calculate accuracy, precision and recall, F1-score for MLP
6.3. Compare the performance of SVM and MLP to assess the clustering quality

## 4. Experiment Results

### 4.1. Computational Pipeline for Subtype Discovery and Classification

The entire process of this study conducted through several sequential steps:

1. Preparation of Gene Expression Data:

Initially, we prepared the gene expression data by cleaning and organizing it to ensure accuracy and consistency for subsequent analysis.

2. Feature Selection Algorithm Application:

In order to calculate the variance measure on the training data, the second step was applying our suggested feature selection algorithm to the complete set of features. Next, we highlighted the gene expressions that had the most discriminative power between the various subtypes of breast cancer by selecting the top-ranked Features. 211 gene expressions were chosen.

3. Graph Representation Using Cosine Similarity:

In the third step, we represented the data as a graph using cosine similarity to capture the relationships between data points better.

4. Clustering with BIGCLAM Algorithm:

The fourth step involved using the BIGCLAM algorithm to cluster the data. These clusters were then used as new labels for our dataset, facilitating supervised classification.

Table 3 displays the number of patients assigned to each class after clustering.

**Table 3:** Number of patients per class after clustering

| Class | Number of Patients |
|---|---|
| 0 | 2807 |
| 1 | 149 |
| 2 | 88 |
| 3 | 140 |
| Total | 3184 |

Table 4 shows how our BIGCLAM-discovered clusters (Class 0–3) overlap with known PAM50 subtypes. These clusters were not constrained to match PAM50 and were generated entirely through unsupervised learning. Notably, each cluster contains a mix of PAM50 labels, suggesting that our method captures different underlying gene expression patterns. For example, Class 0 includes samples from all four PAM50 subtypes, indicating overlapping biological signals not captured by PAM50 alone.

**Table 4:** Distribution of PAM50 Subtypes Across New Clustering Analysis

| PAM50 subtypes | Classes after Clustering | | | |
|---|---|---|---|---|
| | Class 0 | Class 1 | Class 2 | Class 3 |
| LumA | 1507 | 83 | 58 | 61 |
| LumB | 696 | 39 | 13 | 19 |
| Her2 | 293 | 20 | 10 | 25 |
| Basal | 311 | 7 | 7 | 35 |

5. Data Augmentation:

To address class imbalance, we augmented the data as follows:

- Identified the number of members in each class.

- Determined the class with the maximum number of instances.

- Increased the number of instances in other classes to match this maximum count.

This augmented data was primarily used for training and validation, ensuring balanced class representation.

6. Normalization:

The augmented data was then normalized using the Z-score method, which standardized the gene expression values to have a variance of one and a mean of zero.

7. Classification:

Finally, we performed classification using an SVM with an RBF kernel and a 3-layer MLP. The confusion matrices for training, validation, and test data for MLP and SVM are shown in Figures 2 and 3, respectively.

We employed accuracy, sensitivity, specificity, Matthews Correlation Coefficient (MCC), ROC, and AUC [23,24] to assess the performance of our classification models. The definition of these measures is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{11}$$

$$Specificity = \frac{TN}{TN + FP} \tag{12}$$

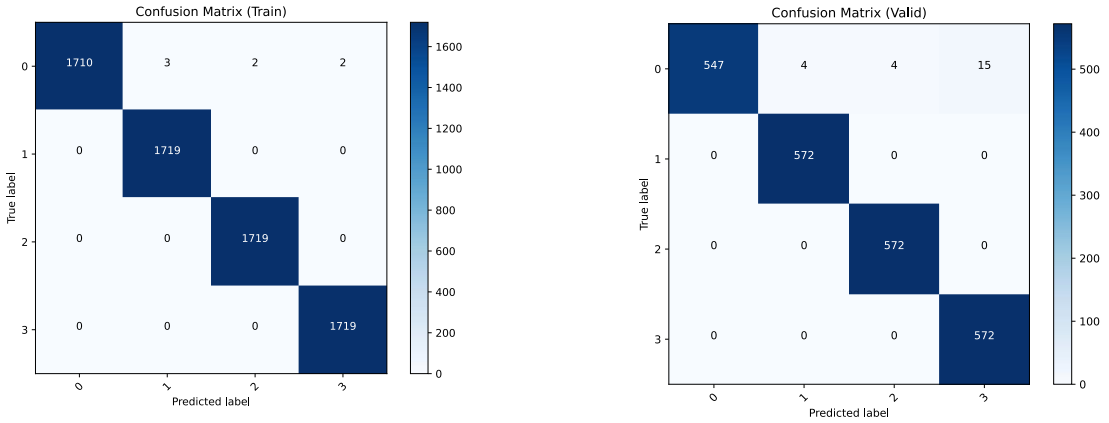
$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP))(TN + FN)}} \tag{13}$$

$$ROC = \frac{TPR}{FPR} \tag{14}$$

$$AUC = \int_0^1 TPR(FPR)dFPR \tag{15}$$

The terms true positives, true negatives, false positives, false negatives, true positive ratio, and false positive ratio, respectively, stand for TP, TN, FP, FN, TPR, and FPR. Table 4 shows the sensitivity, specificity, and MCC for each breast cancer subtype in addition to the values of TP, TN, FP, and FN that were taken from the test confusion matrix. Figures 4 and 5 display the ROC and AUC values for MLP and SVM, respectively. Also, the Performance of the classification regarding sensitivity, specificity, and Matthews correlation coefficient is shown in Table 5.

a. b.




c.

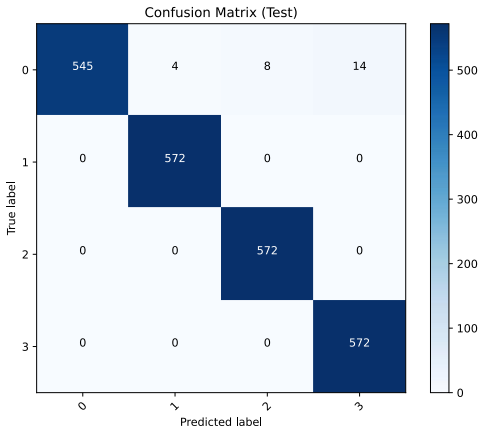
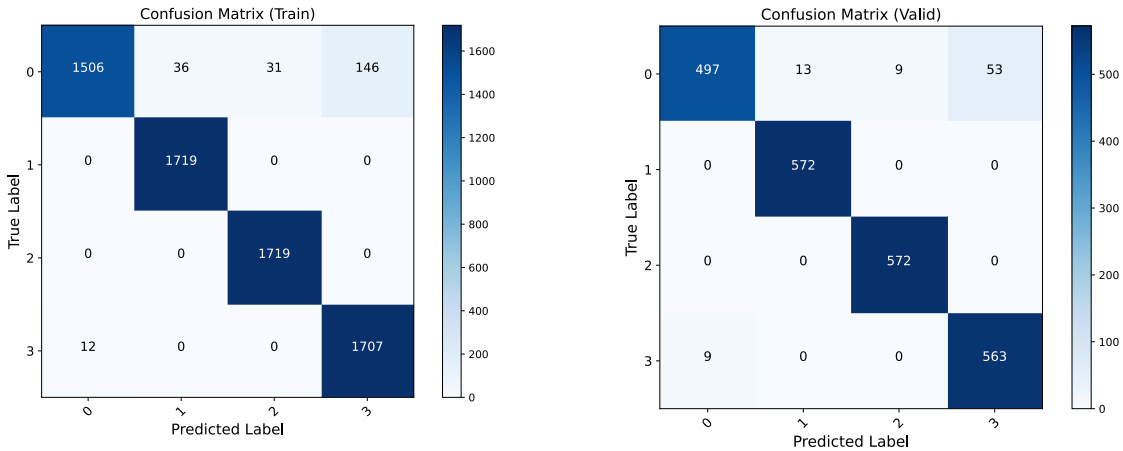



**Figure 2:** Confusion matrices of MLP Classifier **a.** Confusion matrix of train dataset **b.** Confusion matrix of validation dataset and **c.** Confusion matrix of test dataset

a.

b.




c.

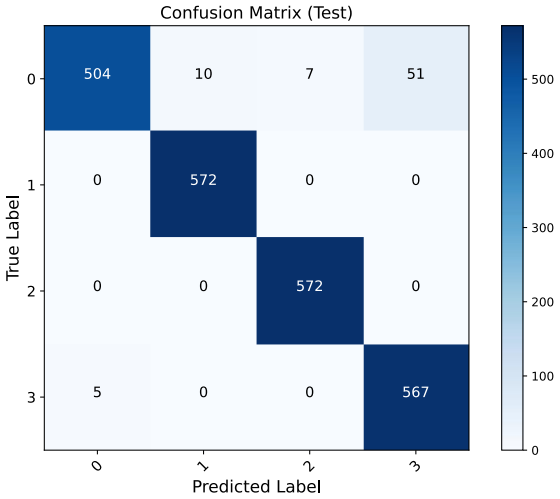
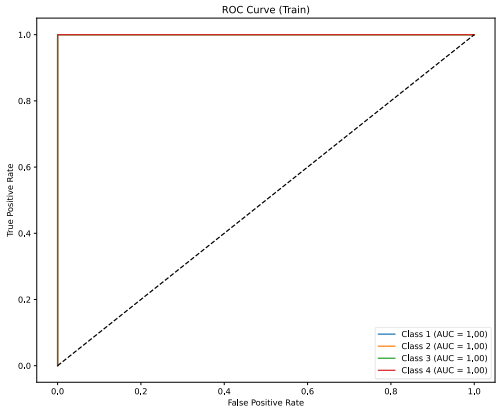
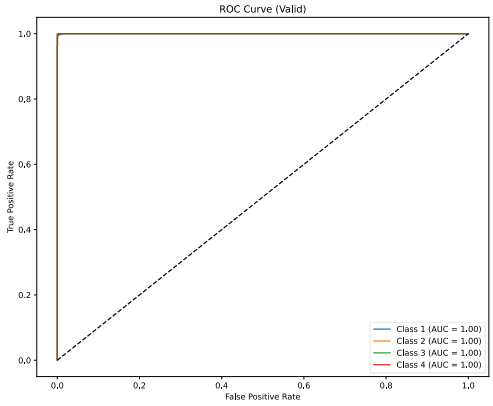



**Figure 3:** Confusion matrices of SVM Classifier **a.** Confusion matrix of train dataset **b.** Confusion matrix of validation dataset and **c.** Confusion matrix of test dataset

a.




b.




c.

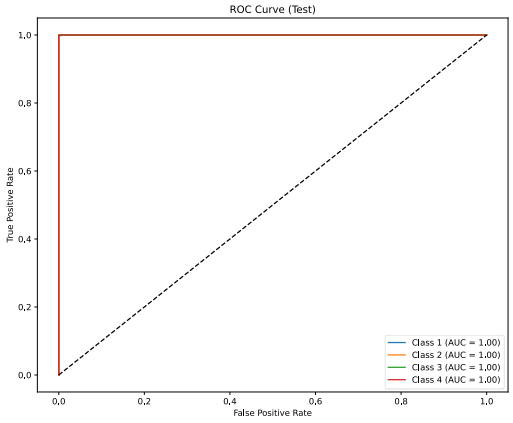


**Figure 4**: ROC and AUC results of MLP Classifier **a.** ROC and AUC results of train dataset **b.** ROC and AUC results of validation dataset and **c.** ROC and AUC results of test dataset

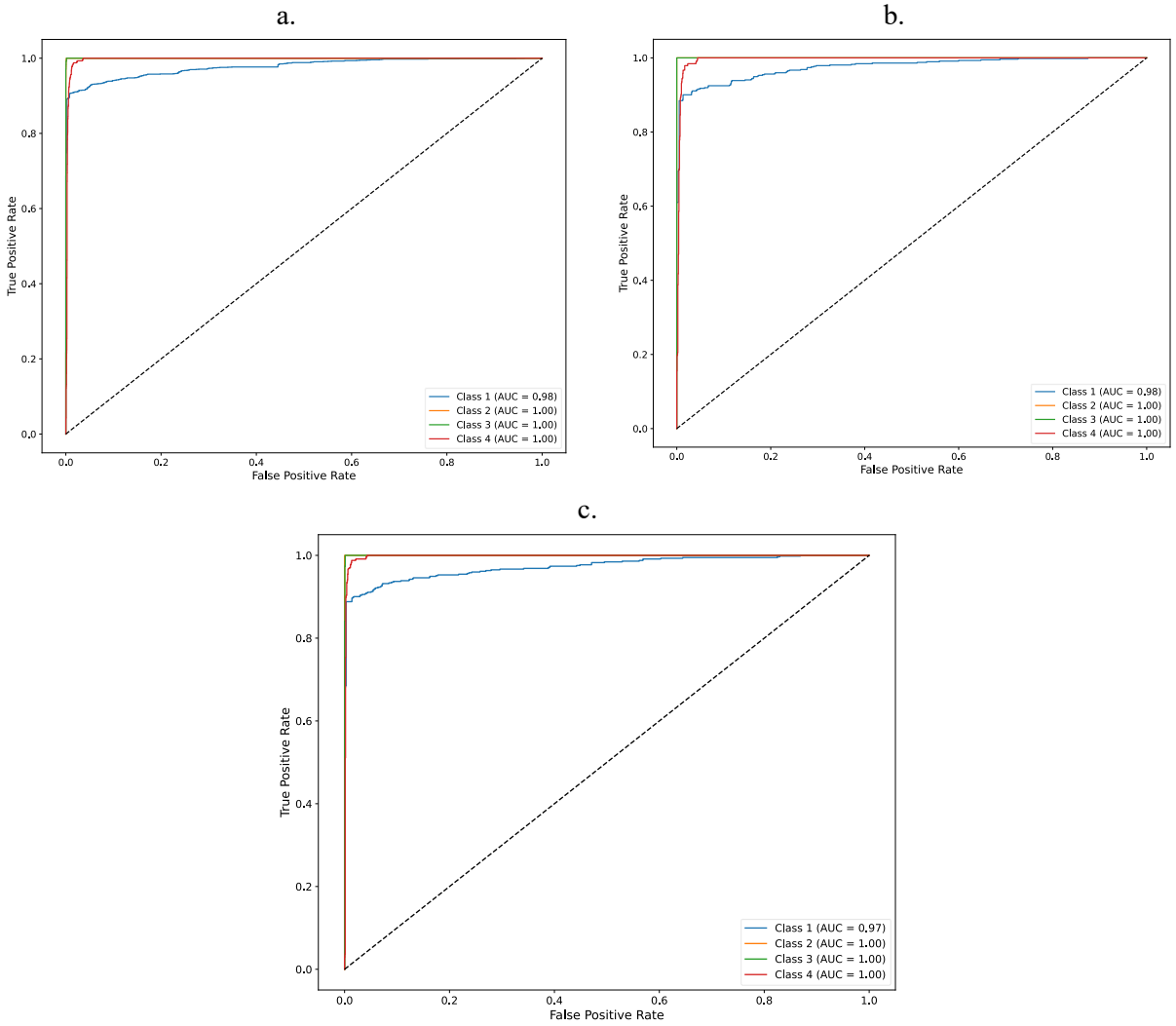



**Figure 5**: ROC and AUC results of SVM Classifier **a.** ROC and AUC results of train dataset **b.** ROC and AUC results of validation dataset and **c.** ROC and AUC results of test dataset

**Table 5:** Performance of the classification regarding sensitivity, specificity, and Matthews correlation coefficient.

| Dataset | Model | TP | TN | FP | FN | Sensitivity | Specificity | MCC |
|---|---|---|---|---|---|---|---|---|
| Train | SVM | 6651 | 20403 | 225 | 225 | 0.97 | 0.99 | 0.96 |
| Valid | SVM | 2204 | 6780 | 84 | 84 | 0.96 | 0.99 | 0.95 |
| Test | SVM | 2215 | 6791 | 73 | 73 | 0.97 | 0.99 | 0.96 |
| Train | MLP | 1719 | 1714 | 3 | 0 | 1.0 | 0.99 | 0.99 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Valid | MLP | 572 | 556 | 4 | 0 | 1.0 | 0.99 | 0.99 |
| Test | MLP | 572 | 550 | 5 | 0 | 1.0 | 0.99 | 0.99 |

To evaluate how informative our discovered subtypes are, we trained MLP and SVM classifiers using them as labels. Table 6 compares the downstream classification accuracy of our model with the reported accuracy of PAM50-based classifiers (AIMS and miniABS). Note that AIMS and miniABS are trained to classify predefined subtypes, while our classifiers use the labels derived from our BIGCLAM clusters.

Our method achieves higher classification accuracy across all clusters, suggesting that the discovered subtypes are more distinguishable and consistent than PAM50 categories. This indicates that BIGCLAM-based clustering captures alternative biological signals that may be more predictive of disease outcomes.

**Table 6:** Comparison of Downstream Classification Accuracy Using Our Discovered Subtypes vs. PAM50 Subtypes(Each cluster is matched to its closest PAM50 subtype for approximate comparison).

| Class | MLP Accuracy(%) | SVM Accuracy (%) | miniABS[25] Accuracy (%) | AIMS[25] Accuracy (%) |
|---|---|---|---|---|
| 0 | NaN | NaN | NaN | 0% |
| 1 vs. LumA | 95.27% | 88.11% | 80.95% | 62.5% |
| 2 vs. LumB | 100.0% | 100.00% | 92.10% | 96.15% |
| 3 vs. Her2 | 100.0% | 100.00% | 75.00% | 47.05% |
| 4 vs. Basal | 100.0% | 99.12% | 100.00% | 93.75% |
| Average | 98.75% | 96.80% | 88.24% | 74.12% |

### 4.2. Biological Significance of Discovered Subtypes

To assess the biological relevance of the discovered clusters, we performed gene-level analysis on the most important features selected for each cluster. Functional enrichment using [DAVID/gProfiler] revealed distinct pathway associations. For instance, Cluster 3 was enriched in immune response and inflammatory signaling pathways, while Cluster 1 showed strong expression of cell cycle-related genes. These patterns suggest that the clusters are not arbitrary groupings but reflect underlying biological mechanisms. This supports the hypothesis that BIGCLAM uncovers meaningful, actionable cancer subtypes beyond traditional PAM50 labels.

## 5. Conclusion

In this study, we introduced a comprehensive approach for categorizing breast cancer subtypes by utilizing gene expression information from the GSE96058 dataset. Our methodology encompassed several key steps: data preparation, feature selection, data representation, clustering, data augmentation, normalization, and classification.

Initially, we prepared and cleaned the gene expression data, ensuring it was suitable for further analysis. We then employed a feature selection algorithm to identify the most discriminative gene expressions, significantly reducing the dimensionality of the dataset while retaining critical information. By representing the data as a graph using cosine similarity, we were able to capture the intricate relationships between data points effectively. The BIGCLAM algorithm was then utilized to cluster the data, providing new labels that facilitated supervised classification.

To address class imbalance, we implemented a data augmentation strategy, ensuring a balanced representation of all classes during training and validation. Following normalization using the Z-score method, we applied two robust classification algorithms: SVM with an RBF kernel and a 3-layer MLP. Our classifiers demonstrated high performance, achieving an average classification accuracy of 98.75%, a sensitivity of 100.0%, a specificity of 99.0%, and a MCC of 99.0% on the test data in MLP classifier and average classification accuracy of 96.8%, a sensitivity of 97.0%, a specificity of 99.0%, and a MCC of 96.0% on the test data in SVM classifier. Additionally, we evaluated our models using ROC curves and the AUC, further validating their effectiveness.

Our findings challenge the adequacy of existing subtyping systems such as PAM50. By uncovering novel clusters directly from gene expression data using BIGCLAM, we identified molecular subtypes that lead to improved classification accuracy. These subtypes appear to capture important biological structure not accounted for by traditional classifiers. Future work should focus on validating these new subtypes in clinical settings and exploring their potential implications for prognosis and treatment.

Future research might include exploring the integrating additional data types, such as clinical and imaging data, to enhance the classification performance further. Additionally, applying our methodology to other types of cancer and diseases could provide valuable insights and contribute to the broader field of personalized medicine.

In conclusion, our study demonstrates the potential of combining advanced feature selection, clustering, and classification techniques to achieve high accuracy in breast cancer subtype classification, paving the way for improved diagnostic and prognostic tools in oncology.

Involving clinical and biology specialists is essential to validate the practical applicability of our findings. Their expertise can ensure that the methods are robust and effective in real-world scenarios, addressing any nuances that may not be fully captured in computational models. This collaboration is key to refining the techniques for use in clinical practice, and its absence is a limitation of our current work.

**Code availability**

The code developed for this research, including implementations of the BIGCLAM method and scripts for clustering breast cancer subtypes using gene expression data, is publicly available in the GitHub repository: Available online: https://github.com/saeedsamimi995/New-Clustering-of-Breast-Cancer-Subtypes-Using-Gene-Expression-Data-and-the-BIGCLAM-Method. The repository includes:Full documentation and workflows for reproducibility, Preprocessing scripts for the GSE96058 dataset and Custom-generated datasets, such as newly assigned cluster labels and intermediate analytical outputs, which are openly accessible in the repository.

**Data availability**

the primary dataset used in this research is the GSE96058 dataset, which is publicly accessible through the Gene Expression Omnibus (GEO) repository, Available online: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE96058, Public on Mar 12, 2018".

the data has been derived from the main dataset, with new labels generated through the research process will be available from the corresponding authors upon request.

---

**References:**


1. Markham, M. J. *et al.* Clinical Cancer Advances 2020: Annual report on progress against cancer from the American Society of Clinical oncology. *J. Clin. Oncol.* **38**, 1081–1101 (2020).

2. Siegel, R. L., Miller, K. D. & Jemal, A. Cancer statistics, 2019. *CA. Cancer J. Clin.* **69**, 7–34 (2019).

3. Metzger-Filho, O. *et al.* Patterns of recurrence and outcome according to breast cancer subtypes in lymph node-negative disease: Results from international breast cancer study group trials VIII and IX. *J. Clin. Oncol.* **31**, 3083–3090 (2013).

4. Perou, C. M. *et al.* Molecular portraits of human breast tumours. *Nat. 2000 4066797* **406**, 747–752 (2000).

5. Sørlie, T. *et al.* Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 10869–10874 (2001).

6. Lowery, A. J., Miller, N., McNeill, R. E. & Kerin, M. J. MicroRNAs as Prognostic Indicators and Therapeutic Targets: Potential Effect on Breast Cancer Management. *Clin. Cancer Res.* **14**, 360–365 (2008).

7. Esquela-Kerscher, A. & Slack, F. J. Oncomirs — microRNAs with a role in cancer. *Nat. Rev. Cancer 2006 64* **6**, 259–269 (2006).

8. Bartel, D. P. MicroRNAs: Target Recognition and Regulatory Functions. *Cell* **136**, 215–233 (2009).

9. Perou, C. M. *et al.* Molecular portraits of human breast tumours. *Nature* **406**, 747–752 (2000).

10. Parker, J. S. *et al.* Supervised Risk Predictor of Breast Cancer Based on Intrinsic Subtypes. *J. Clin. Oncol.* **27**, 1160–1167 (2009).

11. Harbeck, N. *et al.* Breast cancer. *Nat. Rev. Dis. Prim.* **5**, 66 (2019).

12. Paquet, E. R. & Hallett, M. T. Absolute Assignment of Breast Cancer Intrinsic Molecular Subtype. *JNCI J. Natl. Cancer Inst.* **107**, (2015).

13. Seo, M., Paik, S. & Kim, S. An Improved, Assay Platform Agnostic, Absolute Single Sample Breast Cancer Subtype Classifier. *Cancers (Basel).* **12**, 3506 (2020).

14. National Center for Biotechnology Information Gene Expression Omnibus (NCBI GEO), "GSE96058: Gene expression dataset," Available online: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE96058, Published on Mar 12, 2018.

15. Brueffer, C. *et al.* Clinical Value of RNA Sequencing–Based Classifiers for Prediction of the Five Conventional Breast Cancer Biomarkers: A Report From the Population-Based Multicenter Sweden Cancerome Analysis Network—Breast Initiative. *JCO Precis. Oncol.* 1–18 (2018) doi:10.1200/PO.17.00135.

16. *Feature Extraction*. vol. 207 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006).

17. Pedregosa FABIANPEDREGOSA, F. *et al.* Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

18. Virtanen, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat.*

*Methods* **17**, 261–272 (2020).

19. Yang, J. & Leskovec, J. Community-affiliation graph model for overlapping network community detection. *Proc. - IEEE Int. Conf. Data Mining, ICDM* 1170–1175 (2012) doi:10.1109/ICDM.2012.139.

20. Yang, J. & Leskovec, J. Overlapping community detection at scale: A nonnegative matrix factorization approach. *WSDM 2013 - Proc. 6th ACM Int. Conf. Web Search Data Min.* 587–596 (2013) doi:10.1145/2433396.2433471.

21. Cortes, C. & Vapnik, V. Support-vector networks. *Mach. Learn. 1995 203* **20**, 273–297 (1995).

22. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nat. 1986 3236088* **323**, 533–536 (1986).

23. Cichosz, P. Assessing the quality of classification models: Performance measures and evaluation procedures. *Open Eng.* **1**, 132–158 (2011).

24. Janssens, A. C. J. W. & Martens, F. K. Reflection on modern methods: Revisiting the area under the ROC Curve. *Int. J. Epidemiol.* **49**, 1397–1403 (2020).

25. Seo, M. K., Paik, S. & Kim, S. An Improved, Assay Platform Agnostic, Absolute Single Sample Breast Cancer Subtype Classifier. *Cancers 2020, Vol. 12, Page 3506* **12**, 3506 (2020).

**Funding sources**

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

**Author contributions**

Conception and design study: S.Pirmoradi; Formal analysis: S.Samimi; Writing original draft: S.Samimi; Review and editing: all authors, Final Revision: S.Pirmoradi, M.Teshnehlab

**Competing interests**

The authors declare no competing interests.