electronics

MDPI

Article

Identification of Secondary Breast Cancer in Vital Organs through the Integration of Machine Learning and Microarrays

Faisal Riaz [1], Fazeel Abid [1], Ikram Ud Din [2], Byung-Seo Kim [3,*], Ahmad Almogren [4] and Shajara Ul Durar [5]

[1] Department of Information Systems, University of Management and Technology, Lahore 54770, Pakistan; faisalriaz@hotmail.com (F.R.); fazeel.abid@umt.edu.pk (F.A.)
[2] Department of Information Technology, The University of Haripur, Haripur 22620, Pakistan; ikramuddin205@yahoo.com
[3] Department of Software and Communications Engineering, Hongik University, Sejong 30016, Korea
[4] Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11633, Saudi Arabia; ahalmogren@ksu.edu.sa
[5] Management and Organizational Behaviour Business School, University for the Creative Arts, Epsom KT18 5BE, UK; shajara.ul-durar@uca.ac.uk
* Correspondence: jsnbs@hongik.ac.kr

Abstract: Breast cancer includes genetic and environmental factors and is the most prevalent malignancy in women contributing to the pathogenesis and progression of cancer. Breast cancer prognosis metastasizes towards bones, the liver, brain, and lungs, and is the main cause of death in patients. Furthermore, the selection of features and classification is significant in microarray data analysis, which suffers from huge time consumption. To address these issues, this research uniquely integrates machine learning and microarrays to identify secondary breast cancer in vital organs. This work firstly imputes the missing values using K-nearest neighbors and improves the recursive feature elimination with cross-validation (RFECV) using the random forest method. Secondly, the class imbalance is handled by employing K-means synthetic object oversampling technique (SMOTE) to balance minority class and prevent noise. We successfully identified the 16 most essential Entrez gene ids responsible for predicting metastatic locations in the bones, brain, liver, and lungs. Extensive experiments are conducted on NCBI Gene Expression Omnibus GSE14020 and GSE54323 datasets. The proposed methods have handled class imbalance, prevented noise, and appropriately reduced time consumption. Reliable results were obtained on four classification models: decision tree; K-nearest neighbors; random forest; and support vector machine. Results are presented having considered confusion matrices, accuracy, ROC-AUC and PR-AUC, and F1-score.

Keywords: metastasis; microarray; gene expression omnibus; decision trees; random forest; K-nearest neighbours; support vector machine; K-means SMOTE

Citation: Riaz, F.; Abid, F.; Din, I.U.; Kim, B.-S.; Almogren, A.; Durar, S.U. Identification of Secondary Breast Cancer in Vital Organs through the Integration of Machine Learning and Microarrays. *Electronics* **2022**, *11*, 1879. https://doi.org/10.3390/electronics11121879

Academic Editors: Hemant Ghayvat, Sharnil Pandya and Rashid Mehmood

Received: 27 April 2022
Accepted: 9 June 2022
Published: 15 June 2022

1. Introduction

Breast cancer (BC) is the most pervasive cancer in women. Globally, an approximate 19.3 million new cases of cancer were recorded (18.1 million except non-melanoma skin cancer), with nearly 10.0 million deaths from cancer (9.9 million excluding skin cancer non-melanoma) in 2020. BC in women has overtaken lung cancer as the most frequently diagnosed with 2.3 million new cases (11.7%), followed by lung cancer (11.4%), colorectal cancer (10.0%), prostate cancer (7.3%), and stomach cancer (5.6%). BC has been a more significant burden in developing countries due to lifestyle-related risk factors. However, BC incidence rates have recently risen in developed countries due to improvements in health facilities and the acceptance of a westernized lifestyle [1]. About 90% of the deaths caused by BC are due to complications linked to metastasis [2].

BC in Pakistan alone is higher than in any other Asian region with an annual diagnosis of approximately 90,000 new cases and 40,000 of them resulting in death [3]. Approximately

one in nine women are likely to suffer from this type of cancer at some point in their lives. In women over 50 years of age, about 77% of invasive BC occurred but, if diagnosed early, survival rates exceed 90% as presented by the authors in [3]. Young women can also have advanced breast cancer that has a detrimental impact on prognosis. Rural women develop several breast cancers every year in rural areas as it is inherited from mother to daughter. In the number of BC patients worldwide, Pakistan ranks 58th. [4]. According to a recent report, incidence rates of BC are highest in women aged 60–64, however, significant increases in BC rates among women aged 50 to 64 years are projected from 2016 to 2025. In Pakistan alone, the overall estimated BC risk will rise from approximately 23.1% in 2020 to 60.7% in 2025. BC cases diagnosed in younger women aged 30–34 will grow from 70.7% in 2020 to 130.6% in 2025 to [5].

The metastasis in BC patients usually starts with disseminating tumor cells from the primary tumor and their penetration into the bloodstream as a rarely understood process. Circular tumor cells (CTCs) gradually arrest and extravasate through the vascular wall in the capillary beds of distant organs. CTCs inevitably end in the parenchyma leading to secondary site metastatic populations [6]. Furthermore, BC, defined as organ tropism, attacks the bones, lungs, liver, and brain [7]. Metastasized patients with BC have 30–60% bone lesions, 21-32% lung lesions, 4–10% brain lesions, and 15–32% liver lesions [8]. In particular, lung metastases typically appear within five years of BC's primary diagnosis and significantly affects mortality and morbidity. Such metastases interfere with normal lung function leading to coughing, hemoptysis, trouble breathing, and imminent death. An approximate 60–70% of patients who die from BC's lung metastases remain challenging to treat [9]. The prognosis is particularly low for patients with only lung metastases with a median survival of only 25 months [10].

There is much research available identified with cancer genomes. However, most of these have used UCI-free datasets for breast cancer. Moreover, no research has been conducted to accurately reduce microarray gene expression to such a low dimension that features space and highly accurate predictions using the ML and metastatic location, as far as the authors' are aware. This study aims to improve BC patients' life expectancy and quality by identifying the genes responsible for metastasis and the prognosis of metastatic location in vital human organs.

The research work in this paper successfully predicts metastasis's location employing different machine learning algorithms using a dataset that is publicly available named NCBI Gene Expression Omnibus (GEO) GSE14020 [11] and GSE54323 [12]. These microarray datasets are merged to produce a combined dataset with a dimension of $86 \times 20,486$. Microarray technology is a genetic disorder research tool that includes several thousand genetic expressions (features) and hundreds of samples. Each genetic expression calculates the activity level of the genes in a given tissue. Thus, comparing the abnormal cancerous tissue genes offers valuable insights into the disease's pathology and makes it possible to better diagnose future sample estimates as described in [13]. The missing values, high dimensionality, and imbalance class of the gene expression in the dataset are significant when building an accurate breast cancer prediction classifier.

Missing values are imputed using K-nearest neighbors, while the dataset is normalized before proceeding. To overcome the curse of dimensionality, highly correlated features with Pearson correlation $r \geq +0.8$ or $r \leq -0.8$ are removed. The reduced dimensions of the dataset after removing the correlated variables were 86×6602.

To deal with gene expression data and to further reduce the features considered by (a) feature selection methods to determine the most crucial discrimination features and delete irrelevant dependent features, and (b) the feature creation method, which generates new features (low dimensional features) representing the original high dimensional feature in the best possible way.

Recursive Feature Elimination with Cross-Validation (RFECV) using Random Forest for dimension reduction is employed resulting in reduced dimensions of 86×16 (Appendix E).

Whereas class imbalance is handled using a synthetic object oversampling technique (SMOTE) in a novel way by employing K-means SMOTE to balance minority class and prevent noise generation in oversampling.

Lastly, the identification of the 16 most essential Entrez Gene IDs responsible for predicting four different metastatic locations (bones, brain, liver, lungs) on the merged dataset, as mentioned above, with reliable evaluation metrics using classification models such as decision trees, random forest, K-nearest neighbors, and support vector machines.

The rest of the paper is organized as follows: Section 2 describes the material, methods background of the problem under study and Section 3 presents the experimental results and interpretation and methodology. At the same time, Section 4 discusses the research results, inferences, the conclusion obtained, and future work relevant to the study.

2. Materials and Methods

2.1. Background

Cancer treatment needs to recognize metastasis-related molecules and genes and explain these molecules' contribution to the metastatic process. Identifying the genes and molecules related to metastasis and clarifying these molecules' contribution to the metastatic process is vital for cancer treatment [14]. Metastasis transmits tumor cells through the lymph nodes or blood cells from one organ to a remote organ. In the 19th century, Paget questioned whether metastasis development in distant organs was merely a random chance. He reviewed the anatomization of women with BC and discovered a structure of metastatic colonization. He suggested that tumor cells (seed) may have a particular attraction for specific organ microenvironments (soil). This compelling manifestation is known as organotropism. The hypothesis of Paget has now been persistently supported and the significance of tumor cell co-ordination to the microenvironment in promoting the development of metastases is widely recognized. There is a belief that the initial tumor could initiate a pre-metastatic niche before micro metastases are formed, and thus influence the tropism of the organ. However, pre-metastatic niche formation processes are not entirely known [15].

Different subtypes of cellular BC in tissue from the primary breast cancer metastasize the target organ. The metastasis pathway is created by interacting with these subtype cells; the tumor's microenvironment and organ are called the organotrophic metastasis. To achieve remote metastasis, the cancerous cells must first disengage from the primary location and survive as circulating tumor cells (CTCs) without the microenvironment. Most CTCs are removed in a few days from early trapping sites. CTCs that survive and exhaust at a distant organ that forms a micro metastasis may extravasate, generating clinically substantial lesions following a somewhat unforeseeable dormant period that fulfils the division requirements cells in the new microenvironment.

The behavior of organ tropism (lungs, liver, brain and bones) is similar in BC and lung cancer. Nevertheless, they have surprisingly contrasting development times with remote recurrence diagnosed relatively late in BC and early development in lung cancer. In their genetic environment, metastasis is increasingly evident and manifests vital markers of disease. Future clinical outcome is also likely to depend on the characteristics of the metastases [15]. Clinically identified organ-specific metastases indicate that the distant organ site of cancer is not random but somewhat affected by the secondary organ microenvironment. Research has shown that BC cells exhibit organ-specific behaviors for proliferation and migration in the context-specific metastasis locations for BC (brain, lungs, lymph nodes, bones, and liver) [16].

Metastasis is the leading cause of death associated with BC. While the latest treatments have improved significantly, 30–40% of BC patients may ultimately suffer from distant deterioration and surrender to the disease. More than 90% of these patients die from metastasis. These metastasis lesions infiltrate crucial organs and degrade the patient's health, forming several focuses that are challenging to remove surgically and establish resistance to the standard treatments currently available. Therefore, the battle against metastasis is of great importance to winning the war against BC. Therefore, a thorough

understanding of metastasis biology is essential to discover better treatment strategies and achieve long-term therapeutic efficiency [17].

2.2. Breast Cancer Bone Metastasis (BCBoM)

BCBoM is the third most prevalent metastasis location following metastasis in the liver and lungs and usually suggests a provisional diagnosis in cancer patients. If cancer has proliferated to the bones, it can seldom be cured and often delays its progression: BC and prostate cancer cause most skeletal metastases. BCBoM is much more prevalent than primary bone cancers, specifically in adults. After the first BC metastasis, the median survival of patients is 20 months. BCBoMs are significant causes of extreme pain, reduced mobility, pathological fractures, and spinal cord compression. In 10–30% of all cancer patients, pathological fractures occur. BC accounts for 60% of pathological fractures. BCBoM's relative frequency by tumor type is 65–75% in BC patients with advanced metastatic disease. BCBoM is classified into three groups: osteolytic; osteoblastic; and mixed groups. Osteolytic is characterized by average bone loss. The vast majority of BC produces osteolytic metastases. This degradation of the bone is primarily due to osteoclasts and not the direct result of tumor cells. Osteoblastic (or sclerotic) is characterized by new bone deposition or is mixed if a patient has osteolytic and osteoblastic lacerations, or if the metastatic components of a particular individual are osteolytic and osteoblastic, gastrointestinal and squamous cancer are present in BC. While BC mainly gives rise to osteolytic lesions, 15–20% of women suffer from osteoblastic lacerations or both [17].

2.3. Breast Cancer Liver Metastasis (BCLiM)

The liver is a typical metastatic site for cancer. Studies have shown that BCLiM is a complex operation. Factors related not only to BC cells but also to liver microenvironmental factors are involved in this process. Most early metastatic targets in the liver contain few cells, even 12 days after injection of the BC cells. Only a few cells developed into micro-metastatic lesions with patent blood vessels suggesting that lesions that use existing patient blood vessels can thrive in the liver microenvironment while the remaining cells remain dormant in the liver without vascular supplies. However, no clear link has been found between BC subtypes and BCLiM. BCLiM is BC's third most common remote metastatic site compared to the bones, lungs, and brain. As a metastatic site, the liver is observed with clinical and autopsy incidences of 40–50% and 50–62% of all metastatic BCs. Asymptomatic or abdominal distress, ascites, jaundice, abnormal function tests, abdominal pain, and other complications such as sudden liver failure may occur in BCLiM. The median survival period for BCLiM patients is 4–8 months if BCLiM is left untreated. Due to its poor prognosis and limited responsiveness to systemic treatment, BCLiM remains a significant clinical issue [18,19].

2.4. Breast Cancer Brain Metastasis (BCBrM)

A significant series of pathological analyses have shown that breast, colon, lung, and renal cancers are the most commonly identified tumors metastasizing to the brain. The development of tumor cells in the brain's microenvironment stems from cellular transformation processes and genetic propensity, which relies mainly on the interaction between tumor cells and brain-resident cells. This interaction between metastatic tumor cells and the brain's microenvironment expedites colonization [20]. The development of BCBrM is one of the most feared complications after diagnosing advanced BC. BCBrM evolves after the pervasive appearance of metastases in the bones, liver, and lungs. This diagnosis can impact physical function, autonomy, relationships, quality of life, personality and, eventually, self-conception. The tendency to grow BrM for BC varies by subtype. After a BCBrM diagnosis in triple-negative BC, median survival can be as short as five months and 10 to 18 months in other subtypes. Overall, 10–30% of metastatic breast cancer patients experience brain metastases during their illness. However, as with primary BC, the subtype is primordial to metastatic behavior and overall survival. The prevalence of BCBrM for BC is 14% with a median survival of 9–10 months following the occurrence of BCBrM [21].

2.5. Breast Cancer Lung Metastasis (BCLuM)

The lungs are the second most common location for metastatic development with 20–54% pulmonary secondary tumor. BC, colorectal, and renal cancer are the most common predominant laceration leading to BCLuM in adults. In some instances, the cause remains unclear and could be listed as the unknown primary cancer. Pulmonary metastatic disease may have heterogeneous clinical features and may have signs or symptoms with or without them. The BCLuMs are most often associated with endovascular distribution in the distal arterial pulmonary circulation of tumor cells [22]. The lungs would be the first sizeable capillary bed a BC cell faces after it has escaped into the bloodstream. As a CTC in the lungs, they can contact blood vessels of up to 100 m^2. Since these CTCs are five times bigger than the tiny pulmonary capillaries in these capillary beds, the risk of BC cell detention and eventual extravasation into the lungs is high [1]. In reality, about 60% of metastatic BC patients eventually suffer metastases of the bones or lungs in their lives. BC patients are highly vulnerable to BCLuM. Life expectancy is poor with median survival just 22 months after BCLuM treatment. In particular, BCLuM was diagnosed for 60–70% of metastatic BC patients who finally died [23].

2.6. Methodology

In this work, datasets were transformed into a readable format. The selected dataset contains gene expression microarrays of different dimensions that need to be merged based on a common platform (GPL570) and a unique gene identifier (Entrez ID). Data were imputed for missing values and normalized. Gene expression data is known to have the curse of high dimensions. First, highly correlated features were removed and then two-dimensionality reduction techniques were employed to reduce dimensions to cater to this issue. Various classification models were employed with different tests and training split ratios to determine the best accuracy model. In the process, a handful of unique genes that can identify the metastasis location, i.e., brain, bones, lungs, and liver, were able to be identified in a BC patient. The authors of this research have not come across any study that has achieved the said objective. Figure 1 shows the proposed architecture for the identification of breast cancer in vital organs using microarrays.

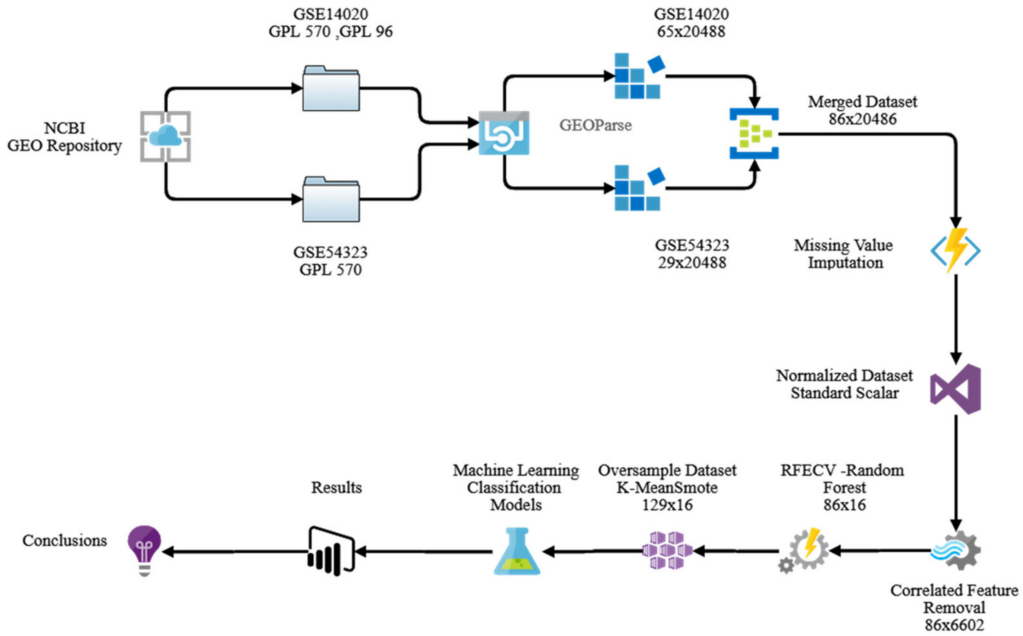

Figure 1. Proposed architecture for identification of breast cancer in vital organs using microarrays.

2.7. Dataset (Dataset Availability)

Two datasets, NCBI-PubMed Gene Expression Omnibus (GEO) GSE14020 [11] and GSE54323 [12], are used in this research. The Gene Expressions Omnibus (GEO) project was launched to increase demands for a public repository of high-performance gene expression data. GEO provides a scalable and open architecture that allows the submission, processing, and testing of heterogeneous data sets from high-performance gene expression and genome hybridization studies. GEO does not plan to substitute internal gene expression databases, which gain from systematic data sets and are structured to simplify a specific analytical approach to function as a tertiary central data delivery center. GEO has three main data entities: samples; platforms; and series. These entities are designed for gene expression and studies on genomic hybridization.

In essence, a platform is a collection of samples that determine what molecules can be identified. A sample defines the collection of examined molecules and refers to a single molecular abundance data platform. A series organizes samples into coherent data sets. The GEO repository is publicly available in [24]. Dataset GSE14020 contains 65 samples collected using two platforms, GPL96 (36 samples with 22,283 gene probe ID) and GPL570 (29 samples with 54,675 gene probe ID) (Appendix A) whereas in dataset GSE54323 29 samples were collected using GPL570 (54675 gene probe ID) (Appendix B) summary in Table 1.

Table 1. Selected Data Sets and Number of Samples.

Dataset	Platform	No. of Samples
GSE14020	GPL96	36
	GPL570	29
GSE54323	GPL570	29

GPL96 is [HG-U133A] an Affymetrix Human Genome U133A array. Human Genome U133 (HG-U133) arrays allow the examination of gene expression across the genome or concentrating on a subset of well-defined genes using single or multi-array plate cartridges. HGU133 is based on the same gene information and identical sample technique to measure the gene expression thoroughly and reliably. GPL570 is [HG-U133_Plus_2] an Affymetrix Human Genome U133 plus 2.0 array. A systematic study of genome-wide expression in a single array, U133 plus 2.0 Array, analyses the relative expression above 47,000 transcript versions with over 38,500 well-known uni-genes and genes. It offers 9900 more samples, representing 6500 more new genes than the previous HG-U133 set with over 54,000 samples and 1,300,000 special features of oligonucleotides [25].

- Entrez Gene

Entrez Gene is a gene-specific database (NCBI National Center for Biotechnology Information) located on the US National Institutes of Health Campus in Bethesda, MD, USA. Entrez Gene produces unique Gene ID (integers) as stable identifiers for a subset of model organisms for genes. It detects and uses such identifiers to incorporate various information including summary descriptions, nomenclature, gene-specific and gene product-specific sequence accessions, pathway and protein interaction reports, chromosomal localization, and related markers phenotypes. Since Gene ID is used in other NCBI databases to describe gene-specific information, the complete Entrez Gene report contains a wealth of links to citations, sequences, variants, homologs, and databases of gene-specific literature beyond NCBI [26].

2.8. Data Pre-Processing

Merging of datasets is inevitable, but the raw data (SOFT files, i.e., simple omnibus format in text) must go through pre-processing before that. The GEOParse package was used in this study to facilitate the researchers in genome studies and download and load the SOFT files from the Gene Expression Omnibus database.

- SOFT Files

The Simple Omnibus Text Format (SOFT) is designed to submit and download data quickly. SOFT is a simple line-based, plain text format that allows SOFT files from regular spreadsheet and database applications. A single SOFT file may contain data tables and descriptive information for samples, series records, and multiple platforms [27].

- GEOParse

GEOParse is a python package used to query and retrieve data from the Gene Expression Omnibus database (GEO) [28]. Salient features of this library are:

Download GEO series datasets as SOFT files.

1. Download supplementary files for the GEO series to use locally.
2. Load GEO SOFT as easy to use and manipulate objects.
3. Prepare data for GEO upload.

In this study, datasets are GSE14020 and GSE54323. All the respective samples from each dataset are downloaded in a data frame (data arranged in tabular format as rows and columns).

2.9. Data Transformation

Dataset GSE14020 contains 65 samples collected using two platforms, GPL96 (36 samples with 22,283 gene probe ID) and GPL570 (29 samples with 54,675 gene probe ID), whereas dataset GSE54323 29 samples were collected using GPL570 (54,675 gene probe ID). Samples are collected using different platforms with different numbers of gene probe IDs. To induce uniformity across the dataset, samples were merged based on the common platform GPL570 (22,283 gene probe ids) and Entrez Gene ID (Unique Gene record identifier), thus reducing the number of features/gene probes IDs to 20,486. The dataset shape achieved after transformation is $86 \times 20,486$ and $X = \{x_1, x_2, x_3, x_4, \ldots\ldots x_n\}$ where $x_1, x_2, x_3, x_4, \ldots\ldots x_n$ are the features/independent variables and $y \equiv$ prognosis location of metastasis (lungs, brain, bones, liver). The distribution of four controlled samples is shown below in Figure 2. The histogram's careful analysis shows that samples are not normally distributed; instead, it is right-skewed. The data, therefore, needs transforming for further processing.

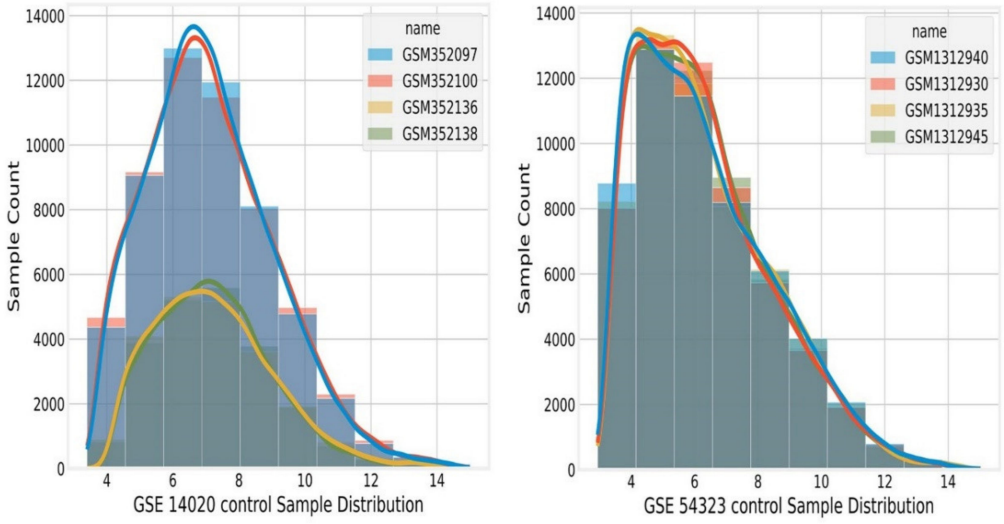

Figure 2. Four Controlled Samples distribution.

- Missing value imputation

The gene expression details from microarray experiments are usually in large matrices of rows (gene expression level) and columns (various experimental environments). Although microarray technology is widely used, the information obtained mostly suffers from missing values. Microarray data may include missing values of up to 10%, and (in

some cases) 90% of genes have one or more missing value in some data sets. Missing values exist for numerous reasons including microarray artifacts, inadequate resolution, hybridization, image noise, and corruption.

Furthermore, suspicious values are also sometimes reported as missing values. The presence of missing values in gene expression may harm subsequent research. Missing values are found to give specific algorithms a non-trivial negative effect. Repetition of experiments can be avoided by imputing missing values. Different algorithms are available for imputing the missing values in gene expression [29].

The imputation process for missing values exploits two data types from the data matrix. The first type is an existing correlation in the data matrix. Since the gene is involved in similar cellular functions, the gene expression data matrix has an identical gene expression profile. A correlation exists in rows and columns under the same genes—similar behavior under similar conditions. The second type is domain expertise of data or the process itself. The experience in the domain is highly beneficial to estimate missing values. Algorithms used for missing value imputation can be categorized as global and local approaches. The global correlation information is extracted based on the entire data matrix, whereas in a local approach only a subset of genes is used to show a gene with a missing value and high correlation value. K-nearest neighbor imputation (KNN impute), the earliest and most well-known imputation algorithm, has been used in this study. KNN impute falls under the category of local approach. Missing values in the dataset are shown in Appendix A, highlighted in red.

- KNN (K-nearest neighbors) impute

This differs from other approaches as it does not work with an actual mathematical model. On the contrary, the inference is performed by comparing new samples with existing ones (defined as instances). KNN is an approach that can be easily employed to solve clustering, classification, missing value imputation, and regression problems. The main idea behind the clustering algorithm is straightforward. Consider a data generating process p_{data} and a finite dataset is drawn from this distribution:

$$X = \{\overline{x_1}, \overline{x_2}, \overline{x_3}, \overline{x_4}. \ldots . \overline{x_n}\} \tag{1}$$

where $\overline{x_i} \in R^N$

$$d_p = (\overline{x_1}, \overline{x_2}) = \left(\Sigma_{j=1}^{N} \left| x_1^{(j)} - x_2^{(j)} \right|^p \right)^{1/p} \tag{2}$$

where $p = 2, d_p$ represents the classical Euclidean distance, which is usually the default choice. In particular cases, it can be helpful to employ other variants such as $p = 1$ (the Manhattan distance) or $p > 2$. Even if all the properties of a metric function remain unchanged, different values of p yield results can be semantically diverse. The KNN algorithm determines the *K* closest samples of each training point. When a new sample is presented, the procedure is repeated with two possible variants:

1. With a predefined value of K, the KNN is computed.
2. With a predefined radius/threshold r, all the neighbors whose distance is less than or equal to the radius are computed.

The philosophy of KNN is that similar samples can share their features. For example, a recommendation system can cluster users using this algorithm and, given a new user, find the most similar ones (based, for example, on the products they bought) to recommend the same category of items. In general, a similarity function is defined as reciprocal of distance (there are some exceptions, such as the cosine similarity):

$$s = (\overline{x_1}, \overline{x_2}) = f\big(d_p(\overline{x_1}, \overline{x_2})\big) = \frac{1}{d_p(\overline{x_1}, \overline{x_2})} \text{for } d_p(\overline{x_1}, \overline{x_2}) \neq 0 \tag{3}$$

Two different users (A and B), who are classified as neighbors, will differ under some viewpoints, but at the same time, they will share some peculiar features. This statement

authorizes one to increase the homogeneity by suggesting the differences. For example, if A liked the book b_1 and B liked b_2, we can recommend b_1 to B and b_2 to A. If this hypothesis were correct, the similarity between A would be increased; otherwise, the two users will move towards other clusters that better represent their behavior [30]. The missing data is used as the test case for the imputation of the missing value. Available and missing features represent input feature space and class label (output). Its K-nearest neighbors from exclusive features are identified whose label imputes the missing attribute [31].

- The KNN-based method

The KNN approach chooses gene expressions identical to the gene, in which missing values can be imputed. In experiment 1, assume gene A has one missing value. This method will classify K additional genes with a value obtained in experiment 1, with an expression identical to A in experiments 2-N (N is total experiments). The missing value in gene A is estimated from the weighted average values of K closest genes. Each gene's contribution is weighted on a weighted average by the gene expression similarity to gene A. After evaluating several matrices for gene similarities such as the Euclidean distance, Pearson correlation, and variance minimization, the Euclidean distance was reasonably accurate. It is pretty surprising provided that the Euclidean distance is sensitive primarily to outliers. Outliers are most likely to exist in microarray data, however, log base2-transformation significantly reduces the outlier's effect on gene similarity [32].

As shown in Figure 3, in this study, K-nearest neighbors are used to input missing values. Each sample's missing values are changed using $K = 10$.

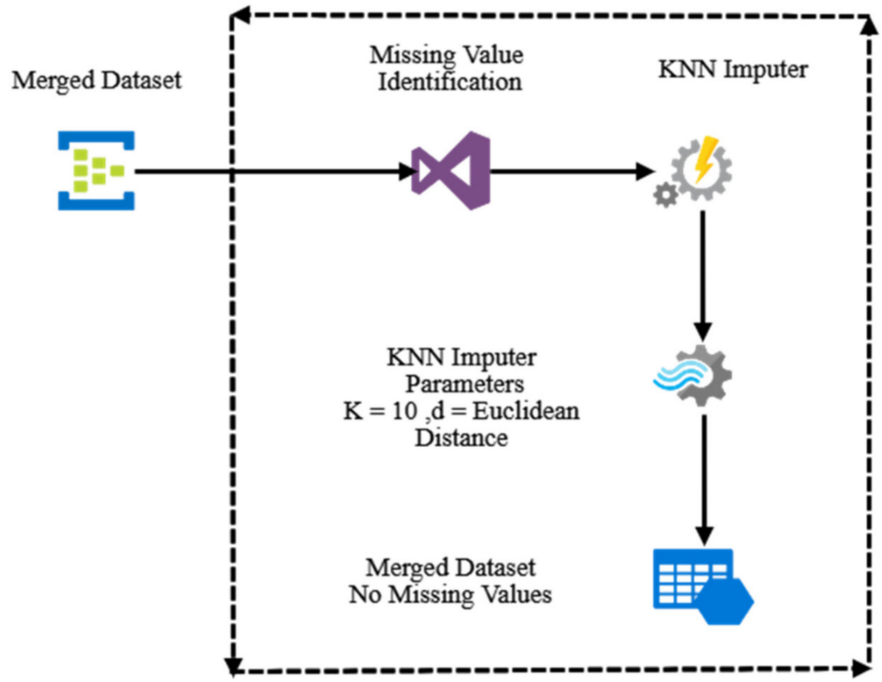

Figure 3. KNN impute Methodology.

($K = 5 - 10$ is a good choice for gene expression arrays missing value imputation in human tumors) [33]. Adjacent neighbors' mean values, the closest neighbor in the training set, and weights by inverse distance. The distance here is the Euclidean distance. More immediate neighbors to a point under query would have a more significant effect than the farthest neighbors in this scenario. The two samples are close if the features that neither is missing are close [34].

- Data Normalization

Generally, the feature values range varies widely in various databases. When feature values vary, several training algorithms' objective function does not operate correctly.

Assume that the algorithm's objective function uses the distance between two features; the feature controls the distance with an extensive range of values that deceive the objective function. Similarly, the gradient-based back-propagation algorithm performs better if the attribute values are of the same range. The features are scaled to a specified range to eliminate one factor's influence over another and faster convergence. Data normalization is the process by which feature values are scaled to a specified range. In microarray data, the value of the attributes ranges from a low to an enormous value. This paper has already shown the sample distribution in Figure 2, which indicates that the data needs to be normalized. Therefore, data normalization is unavoidable for the microarray dataset before applying any training algorithm. Here, the standard scaler normalization method is applied to scale the data [35]. Independent variables or features are standardized to zero mean and scaling to unit variance. The standard score is calculated as:

$$Z_i = \frac{x_i - \mu}{\sigma} \tag{4}$$

- Where mean: $\mu = \frac{1}{N}\Sigma_{i=1}^{N}(x_i)$ and standard deviation:

$$\sigma = \sqrt{\frac{1}{N}\Sigma_{i=1}^{N}(x_i - \mu)^2} \tag{5}$$

The normalized dataset after merging is shown in Appendix C.

- Removal of Correlated Variable

A multivariable analysis is a widely used statistical tool in medical research if the correlation of several predictive variables with study measurements is calculated. However, the multivariable efficiency of analysis depends on the correlation structure between predictive variables. Moreover, the multivariable efficiency of analysis also depends on the correlation structure between predictive variables. The multivariate analysis assumes that all predictive variables are not correlated. Multi-collinearity or issues arise when the model's covariate is not independent. Consequently, it leads to biased coefficient estimation and loss of power in genomics and medicine studies. Statistically, correlation assesses a linear association among two continuous variables.

Correlation is measured by the correlation coefficient, representing the strength of linear association between the variables. A correlation coefficient of zero means that two continuous variables have no linear relationship and a correlation coefficient of -1 or $+1$ reveals an entirely linear relationship. The higher the correlation, the closer the correlation coefficient gets to ± 1. The variables are positively related if the coefficient is a positive number and the variables are inversely related if the coefficient is a negative number. There are two key correlation coefficients, i.e., Pearson's correlation and Spearman's rank correlation coefficient. The proper application of the correlation coefficient form depends on the type of variables under study [36]. In the study, only Pearson's Correlation coefficient is being considered. Pearson's correlation coefficient is denoted as ϱ for a population parameter and as r for a sample statistic. Pearson's correlation between variables x and y is given by:

$$r = \frac{\Sigma_{i=1}^{n}(x_i - x)(y_i - y)}{\sqrt{\left[\Sigma_{i=1}^{n}(x_i - \overline{x})^2\right]\left[\Sigma_{i=1}^{n}(y_i - \overline{y})^2\right]}} \tag{6}$$

Before proceeding further, the highly correlated features were removed. All the independent variables having a Pearson correlation coefficient greater than 0.8 or less than -0.8 were removed. 13,884 features were removed, thus reducing the reduced dimension of the dataset to 86 $\times$ 6602 Appendix D.

- Dimensionality Reduction

Even after removing highly correlated features, the dimensions of the dataset were still too large to handle. Next, the recursive feature elimination technique was used to

reduce the dataset's dimension further. Moreover, RFE (Recursive Feature Elimination) will identify the most robust predictor/features.

- Recursive Feature Elimination (RFE)

Random forest (RF) is a supervised machine learning algorithm. It generally works well with high dimensional datasets and can recognize a given result's strong predictors without making any basic model assumptions. However, correlated predictors are a common problem with high-dimensional data sets. RF's efficiency in recognizing the most potent predictors decreases the significant calculated scores of correlated variables. Highly correlated variables were already removed in the last step. The Random-Forest-Recursive Feature Elimination (RF-RFE) algorithm is a proposed solution. RFE performs the selection of features by iteratively training a model, classifying features, and eliminating the lowest ranking attributes [37]. RFE needs a range of features to be preserved. However, the number of features that are authentic is also not decided in advance. Cross-validation is used with RFE to score specific feature subsets and select the best scoring collection of possible features or attributes. In this experimental study, the Stratified K-fold cross-validation was used for recursive feature elimination. Stratified K-Fold shuffles the data split into n_splits parts. Each split is used as a test set. It constantly shuffles data once before splitting and does not overlap test sets.

Recursive Feature Elimination with Random forest with stratified K-fold cross-validation reduced the optimal number of features from 6602 to 16. The optimal number of features and their importance is shown in Figures 4 and 5 below.

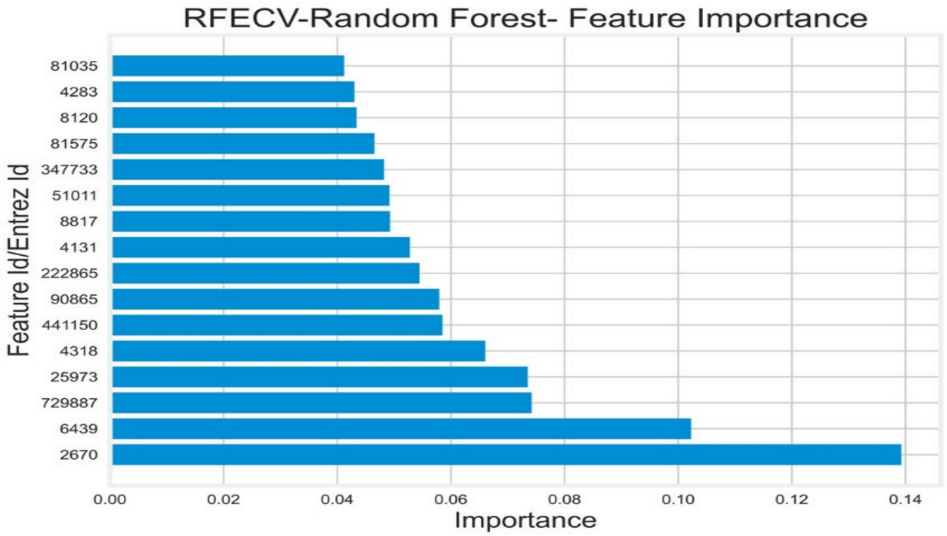

Figure 4. RFECV-Random Forest.

- Class imbalance

The critical problem in microarray data analysis is a limited sample size with high dimensionality. Class imbalances compound this situation. Data imbalance concerning multiclass classification has been recognized as a challenging problem for machine learning techniques as it directly impacts the classification model's performance. Most of the machine learning classification algorithms assume the classes to be balanced. As a result, the algorithm would favor the majority class and ignore the minority classes leading to poor classification models, leading instantiation, and poor performance metrics. The class imbalance will reduce the credibility of the accuracy of classification. In addition, notable features with an imbalanced data set are often problematic as they are not evenly distributed in the training set [38].

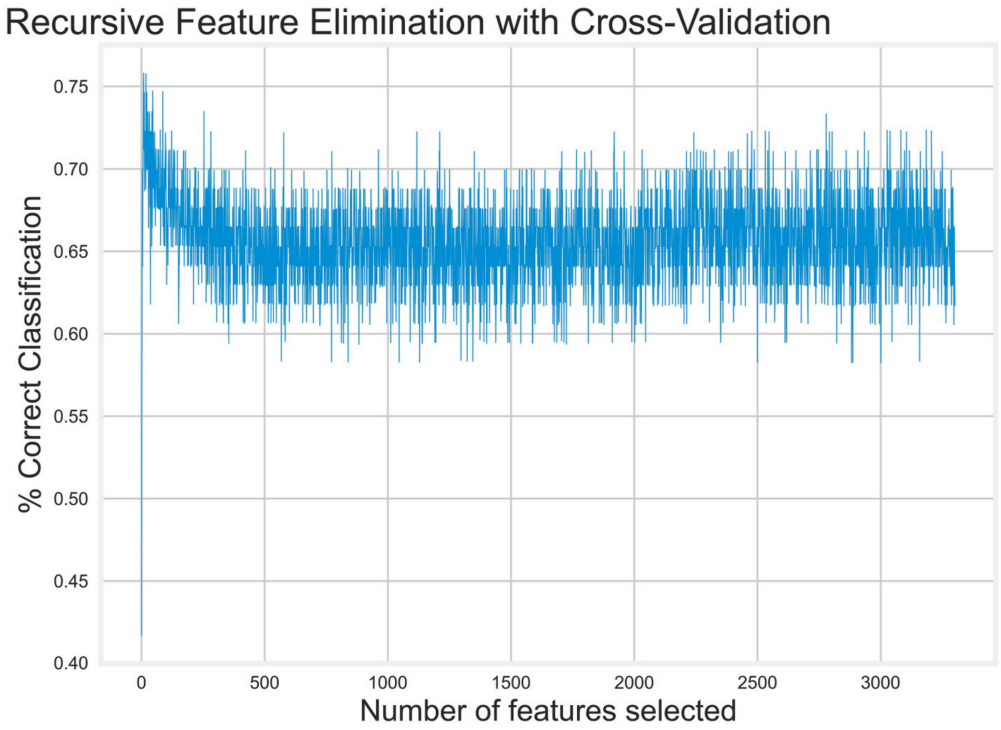

Figure 5. Feature Importance.

In the following datasets is the status of class imbalance in two datasets, GSE14020 and GSE54323. Figures 6 and 7 show the imbalanced nature of the two datasets.

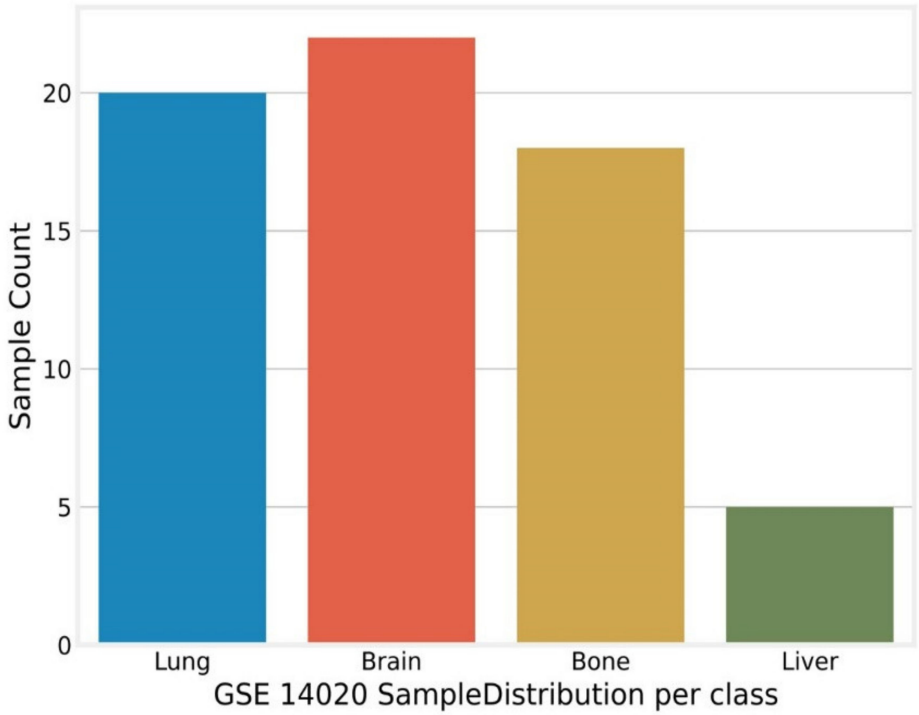

Figure 6. GSE14020 Class imbalance.

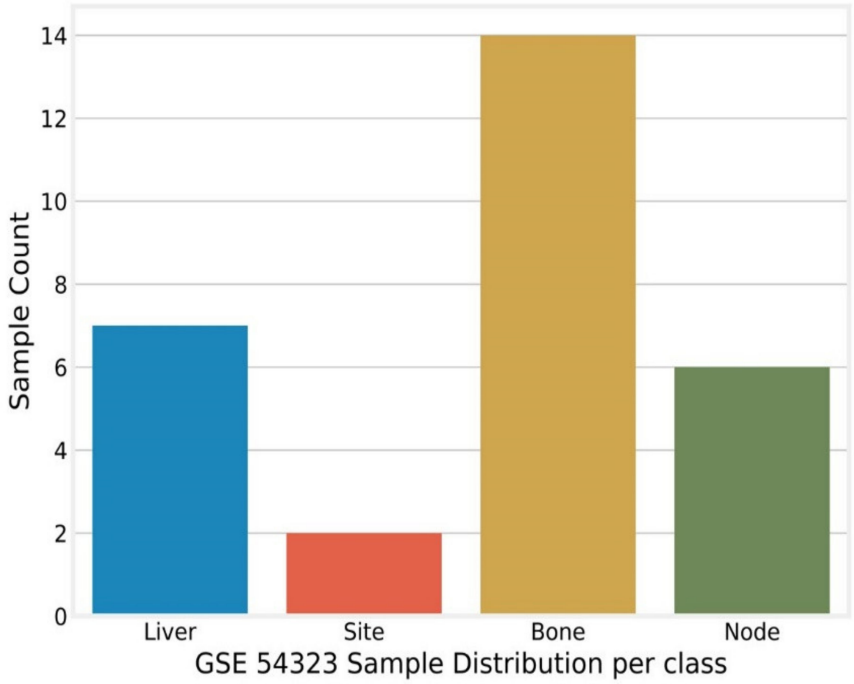

Figure 7. GSE54323 Class imbalance.

After the merging of two datasets, The overall class imbalance situation of a dataset is shown in Figure 8.

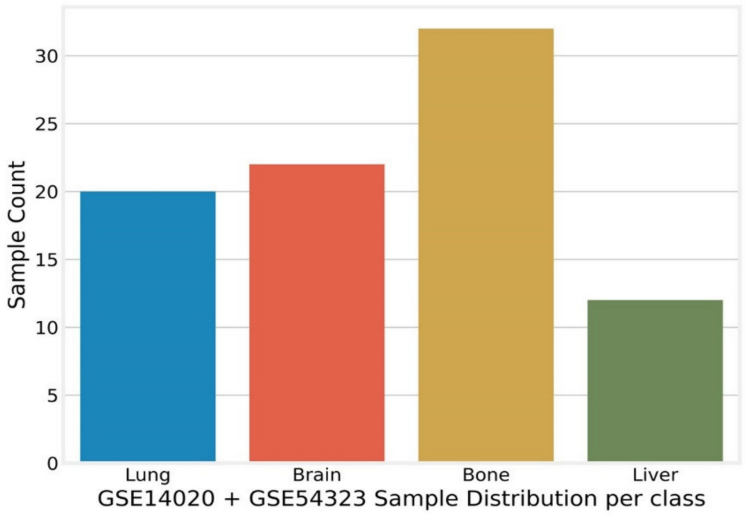

Figure 8. Merged Dataset Class imbalance.

The number of observations for site (sub-cutaneous site) and node (Lymph Node) were not significant. These samples were dropped before proceeding further, as shown in Figure 7.

- SMOTE

Over time, various resampling techniques have emerged to cater to the class imbalance problem. Frequently used methods are over-sampling and under-sampling. The underlying principle of these methods is to randomly remove samples or randomly pick samples from the minority and replicate them, causing either information loss or overfitting. SMOTE is a prevalent resampling technique used in imbalance classification datasets. SMOTE (synthetic minority oversampling technique) is an approach that oversamples the minority class by creating synthetic examples rather than creating over-sampling with replacement. Synthetic

samples are explicitly generated by acting in the feature space instead of the data space. The minority classes are oversampled by selecting each minority class sample and inducing synthetic samples along the line, joining any minority class nearest neighbors. Nearest neighbors are randomly chosen based on the aggregate of over-sampling required. For example, suppose the aggregate of over-sampling needed is 300%. In that case, only three neighbors are selected (based on five nearest neighbors) and only one sample is created synthetically in each direction. New synthesized samples are created as the difference between the feature vectors, i.e., the sample being considered, and the nearest neighbor associated with it. This difference is multiplied by an arbitrary value between zero and one and then added to the feature vector considered. Consequently, it causes the arbitrary point to be selected along the line segment between two specific features. Therefore, this method dictates the decision region of the minority class towards more generalization [39].

The SMOTE samples are two similar linearly combined samples of minority class $\left(x \text{ and } x^R\right)$ and are defined as:

$$S = x + \mu \cdot \left(x^R - x\right) \tag{7}$$

$0 \leq \mu \leq 1$; x^R is randomly chosen among the five-minority class nearest neighbors of x. In this study, K-means SMOTE [40] has been used. K-means SMOTE assists classification by generating minority class samples in safe and crucial areas of the input space. The technique prevents noise generation and overcomes imbalances between and within classes effectively. K-means SMOTE works in the following steps:

- Use the K-means cluster algorithm to cluster entire data.
- Choose clusters with a significant number of minority class samples.
- Assign more synthetic samples to clusters with sparse distribution of minority class samples.

Figure 9 depicts the balanced data samples in each class after applying the K-means SMOTE oversampling technique.

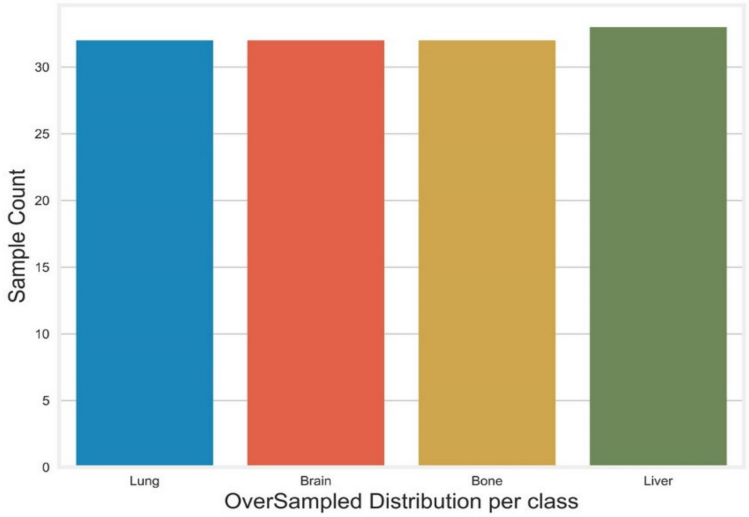

Figure 9. Balanced Dataset (KMeans-Smote).

Since the data from 86 × 16 to 129 × 16 has been oversampled, the dataset's final shape is shown in Appendix F [41].

- Sampling

In this study, the Stratified shuffle split was sued for sampling. Stratified sampling tends to split a data set so that each split is identical to something. A classification setting is always chosen to ensure that training and test sets have roughly the same samples for each target class as the complete set. A 70/30 ratio was employed, i.e., splitting the dataset such that 70% of the samples are reserved for training the model and 30% of the samples for

model validation. The dimension of the training dataset was 90 × 16 and the test dataset was 39 × 16.

2.10. Classification Models

Four different models have been trained based on different classification techniques used for multiclass classification. A brief description of these algorithms is shown below.

- K-nearest neighbor (KNN)

KNN is used for both regression and classification problems. It belongs to the family of supervised learning algorithms. It uses feature similarity to predict the class y for a new set of observations x. It has extensive pattern recognition and classification application and uses information about the neighboring points to classify output labels. KNN classifier is an example-based learning and non-parametric algorithm. It performs well even if the data is non-Gaussian. This algorithm does not learn the model, instead, it learns the training instances that are the foundation of information during the prediction phase. This algorithm uses k distance estimates based on input features. The optimal value of k = 3 is used in this research. However, non-relevant features significantly reduce the accuracy or precision of KNN, even with highly efficient classification. KNN has the downside of not being computationally efficient since it stores the whole training data in memory. Prediction of every new set of observations is required to run down through the full dataset, making it a lazy learning algorithm [42,43].

- Decision Trees (DTs)

DTs are supervised learning and non-parametric methods used extensively for regression and classification. DTs predict the target variable y by learning straightforward decision rules inferred from input features x. A DT is a piecewise constant approximation that breaks down the training dataset into smaller sets with simple if-then-else decision rules. The outcome is a tree with decision nodes. DT classifiers build decision trees for a set of training data. The classifier frequently visits all decision nodes and chooses active splits until a leaf is pure and no further splits are obtainable. Several methods are available to quantify the purity of decision nodes, but the Gini impurity criterion has been employed in this research. DTs can create complex trees that do not generalize well, resulting in an over-fitted model. This problem can be avoided using the maximum depth of the tree. In this research max_depth = 3 [44] has been used.

- Random Forests (RF)

RF is a supervised and non-parametric learning algorithm. RF is suitable for both regression and classification. However, its main application is in classification. RF is an ensemble learning method that is superior to a DT as it mitigates the over-fitting and minimizes the influence of outliers on predictions. Random forests are merely a collection of DTs. In RF, each tree is different from one another. A single tree might be good at predicting, but most likely be overfitting on another part of data. The overfitting can be reduced by averaging the results of multiple decision trees, while retaining the predictive power. Random forest employs the same strategy by inducing randomness to ensure that each tree is different and distinct. In order to build a tree, a bootstrap sample of data is taken. This process is repeated, thus creating a dataset as extensive as the original training dataset. However, RF selects a subset of random features and looks for the best possible test involving those features. These subsets of features are repeated on each node so that each node in a tree can be decided using a different subset of features. In this study, the maximum depth of the tree, max_depth = 2, has been used and the criterion to determine the purity of the node is Gini. In regression tasks, results are the average for prediction. A voting mechanism is used in classification making a soft prediction with the probability for each possible output label. The probabilities predicted by all the trees are averaged, predicting the label with the highest predicted probability [45,46].

- Support Vector Machine (SVM)

SVM is the most popular and extensively used machine learning algorithm for classification problems. SVM predicts the target labels by creating a decision boundary between classes using single or multiple feature vectors. A decision boundary or hyperplane is a line that splits the input variable space by its class. The margin is the distance between the points that lie closest to the line. The hyperplane with maximal-margin is an optimum line that separates the classes with the most significant margin. The vertical (i.e., perpendicular) distance from these closest points to the hyperplane is the most relevant point in defining the classifier's hyperplane and construction. These points are the support vectors that define the hyperplane. SVM was initially proposed to construct a linear classifier, however, a brilliant trick for SVM is a kernel function that enhances the capability to model non-linear higher dimension models. Kernel function adds dimension to input data and makes a non-linear problem to a linear problem in a higher dimension; it calculates the scalar product between two data points in a higher dimension space without explicit mapping from the input data to higher dimensions. There are three types of kernel functions used in SVM: polynomial; linear; and radial. This research study uses Grid Search CV (Grid Search Cross Validation) to find the optimal hyperparameter for its model. Kernel = Linear, C = 1, and gamma = 0.1, where C is the L2 regularization parameter and gamma the kernel coefficient [47]. The above models were trained (70% data) and validated on the test (30% data) dataset.

A comprehensive summary of each classification model is presented in Table 2.

Table 2. Pros and Cons summary of Proposed Classification Models.

Algorithm	Pros	Cons
KNN	Very easy to understand and implement. Does not make any assumption about data. It changes to accommodate the new data points when exposed to new data.	Lazy learning algorithm. Poor performance with high dimension dataset. Data scaling is required.
DT	Data scaling or normalization is not required. Missing values does not have considerable impact. Easy to interpret and visualize.	Prone to overfitting the model. Training time is higher. Sensitivity to data changes is quite high. A small change can affect the result significantly.
RF	Random Forest is an ensemble based on decision trees. It ensures the reduction in overall variance and error. Performs well with higher dimensions. Missing values and outliers do not have considerable impact. Not prone to overfitting.	Not easy to interpret. Tuning of hyperparameters is required to improve performance. Training time is lower but prediction time is higher.
SVM	Provides high accuracy and performance in higher dimensional data. Most suitable algorithm when classes are separable either linear or non-linear. Susceptibility to outliers is low.	Execution time is higher for larger dataset. Performance degrades in case of non-separable classes. Hyperparameter optimization is required for better generalized performance.

2.11. Classification Evaluation Metrics

In this study a multiclass classification problem is being dealt with. In classification problems, accuracy alone is not an excellent metric to validate a classifier. There are different performance metrics available for classifier evaluation. Classification model performance measures used in the study are:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

$$\text{Recall/Sensitivity} = \frac{TP}{TP + FN} \tag{10}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{11}$$

$$\text{F1 score} = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \tag{12}$$

$$\text{True Positive Rate (TPR)} = \text{Sensitivity} \tag{13}$$

$$\text{False Positive Rate (FPR)} = 1 - \text{Sensitivity} \tag{14}$$

All the classifier models have been evaluated based on each classification model's matrices and respective performance. Moreover, the models were compared based on ROC–AUC (receiver operator characteristic–curve area under the curve) and PR–AUC (precisionrecall—area under the curve) using Yellowbrick. AUC is the degree of separability, signifying the ability of the classifier to classify correctly.

3. Results and Discussion

Dataset GSE14020 contains 65 samples collected using two platforms, GPL96 (36 samples with 22,283 gene probe ID) and GPL570 (29 samples with 54,675 gene probe ID) whereas for dataset GSE54323 29 samples were collected using GPL570 (54,675 gene probe ID). Samples were collected using different platforms with different numbers of gene probe IDs. To induce uniformity across the dataset, samples were merged based on common platform GPL96 (22,283 gene probe IDs) and Entrez_Gene_ID (Unique Gene record identifier), thus reducing the number of features or genes probes IDs to 20,486. The dataset shape achieved after transformation was $86 \times 20,486$.

Furthermore, after removing highly correlated variables/features and reducing the dimensionality of the dataset to 86×16, the dataset was oversampled using K-means SMOTE to balance class distribution. The final dimensions of the dataset were 129×16. Pearson correlation was calculated for each independent variable concerning each other as denoted in Figure 10. All the features with negative correlation values are marked red, whereas positive correlations are black.

	222865	25973	2670	347733	4131	4283	4318	441150	51011	6439	729887	81035	8120	81575	8817	90865
222865	1.00	0.56	0.50	-0.09	0.23	-0.21	-0.08	0.58	0.40	0.51	0.40	0.08	0.47	0.22	0.47	0.02
25973	0.56	1.00	0.61	-0.12	0.36	-0.32	-0.10	0.58	0.67	0.33	0.49	0.16	0.45	0.43	0.43	-0.12
2670	0.50	0.61	1.00	0.27	0.68	-0.44	-0.01	0.51	0.58	0.13	0.47	0.22	0.46	0.59	0.46	-0.21
347733	-0.09	-0.12	0.27	1.00	0.36	0.04	-0.15	0.08	0.22	-0.12	0.09	-0.08	0.23	0.18	-0.08	0.00
4131	0.23	0.36	0.68	0.36	1.00	-0.29	0.17	0.34	0.48	-0.02	0.32	0.33	0.38	0.67	0.46	-0.30
4283	-0.21	-0.32	-0.44	0.04	-0.29	1.00	-0.17	-0.20	-0.15	0.14	-0.26	-0.35	-0.36	-0.47	-0.35	0.53
4318	-0.08	-0.10	-0.01	-0.15	0.17	-0.17	1.00	-0.08	-0.14	0.04	-0.22	0.56	0.24	0.27	0.35	-0.16
441150	0.58	0.58	0.51	0.08	0.34	-0.20	-0.08	1.00	0.63	0.38	0.72	0.20	0.45	0.48	0.51	-0.08
51011	0.40	0.67	0.58	0.22	0.48	-0.15	-0.14	0.63	1.00	0.31	0.58	0.05	0.40	0.49	0.42	-0.05
6439	0.51	0.33	0.13	-0.12	-0.02	0.14	0.04	0.38	0.31	1.00	0.17	0.19	0.30	0.08	0.40	0.27
729887	0.40	0.49	0.47	0.09	0.32	-0.26	-0.22	0.72	0.58	0.17	1.00	-0.02	0.30	0.44	0.30	-0.11
81035	0.08	0.16	0.22	-0.08	0.33	-0.35	0.56	0.20	0.05	0.19	-0.02	1.00	0.32	0.51	0.49	-0.20
8120	0.47	0.45	0.46	0.23	0.38	-0.36	0.24	0.45	0.40	0.30	0.30	0.32	1.00	0.48	0.52	-0.25
81575	0.22	0.43	0.59	0.18	0.67	-0.47	0.27	0.48	0.49	0.08	0.44	0.51	0.48	1.00	0.54	-0.37
8817	0.47	0.43	0.46	-0.08	0.46	-0.35	0.35	0.51	0.42	0.40	0.30	0.49	0.52	0.54	1.00	-0.14
90865	0.02	-0.12	-0.21	0.00	-0.30	0.53	-0.16	-0.08	-0.05	0.27	-0.11	-0.20	-0.25	-0.37	-0.14	1.00

Figure 10. Pearson Correlation among features. All the features with negative correlation values are marked red, whereas positive correlations are black.

A heatmap of positive and negative correlated features is shown in Figures 11 and 12.

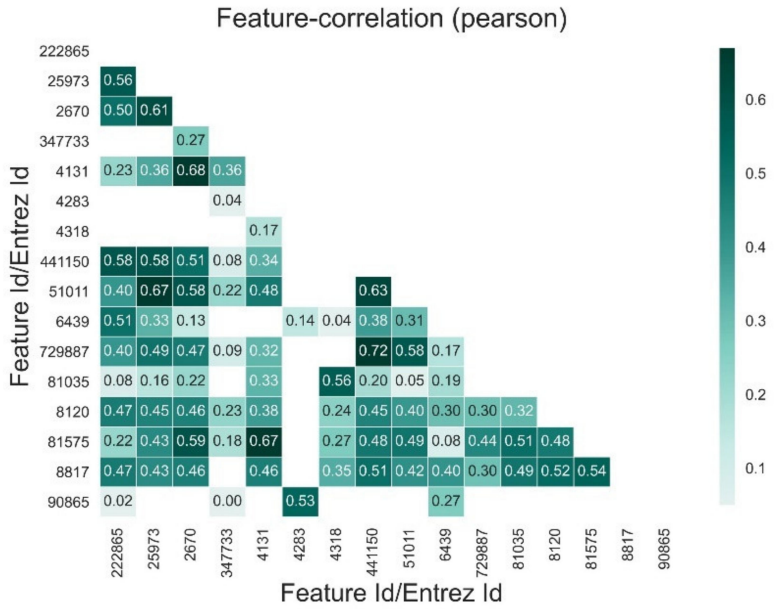

Figure 11. Positive Feature Correlation Heat Map.

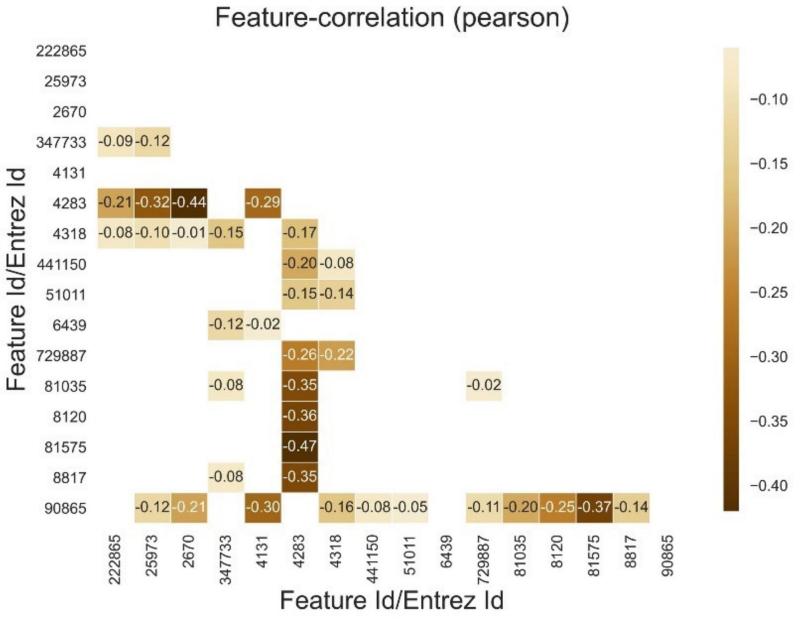

Figure 12. Negative Feature Correlation Heat Map.

This clearly shows that most features exhibit positive correlation but a few are negatively correlated. Analysis of the correlation table in the figures above reveals that 18% of values are strongly correlated $(\pm 0.5 \rightarrow \pm 1.0)$, 34% are moderately correlated $(\pm 0.3 \rightarrow \pm 0.49)$ and 48% are in weak correlation (± 0.29).

Four different classifiers were trained on 70% train, 30% test ratio and evaluated each classifier concerning the accuracy, precision, recall, ROC–AUC, PR–AUC, and F1score. The experiments were evaluated on a virtual machine (VM) with an Intel Xeon CPU ES-2690V4 @2.60 GHz having 12 vCPU and 24 GB of RAM.

3.1. Decision Tree Classifier

Decision tree classifier had a training accuracy of 92% and validation/test accuracy of 87%. Confusion matrix is shown in Figure 13, which indicates the misclassified samples in lungs, brain, and bones class.

Figure 13. Confusion Matrix (Decision Tree Classifier).

Out of 39 total samples, this classifier has misclassified five samples across all four classes. One lungs sample is classified as bone and two as liver, one brain sample is classified as lungs, and one bone sample as liver. The DT classification report is shown in Figure 14. Class liver and lung have low precision of 0.77 and 0.87, respectively, whereas lung has poor recall of 0.78, thus reducing the overall F1 Score for the liver and lungs class.

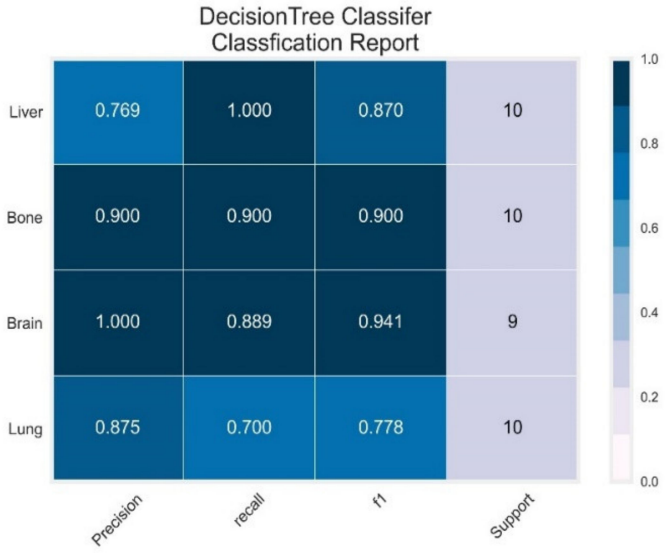

Figure 14. Classification Report (Decision Tree Classifier).

The precision-recall curve for DT classifier is depicted in Figure 15. Precision-recall curve is a metric to evaluate the quality of a classifier. It shows the trade-off between precision and recalls for each class. A larger area under the curve represents the classifier with high precision and recalls the best case scenario with an average precision of the classifier model.

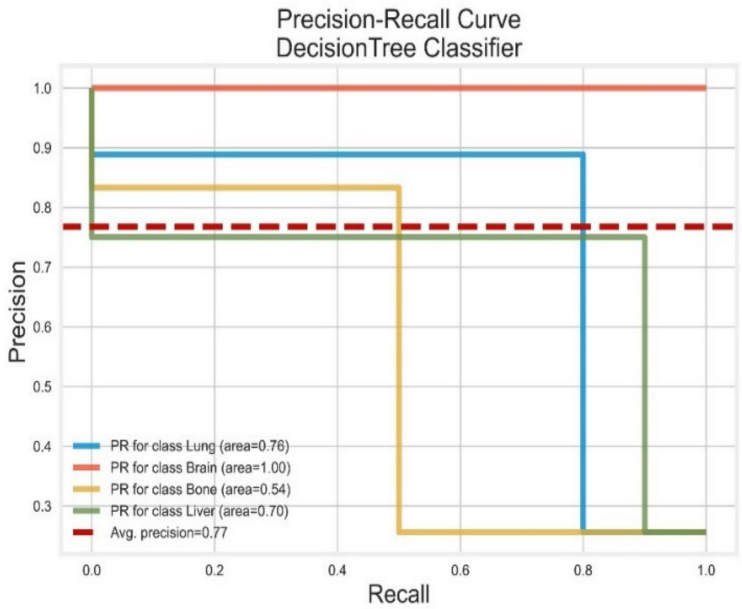

Figure 15. Precision-Recall Curve (Decision Tree Classifier).

The DT classifier had an average precision of 0.77, showing brain class with the highest PR–AUC = 1.0, and bones with the lowest PR–AUC = 0.54. The Receiver Operator Characteristic (ROC) is shown for the DT classifier in Figure 16.

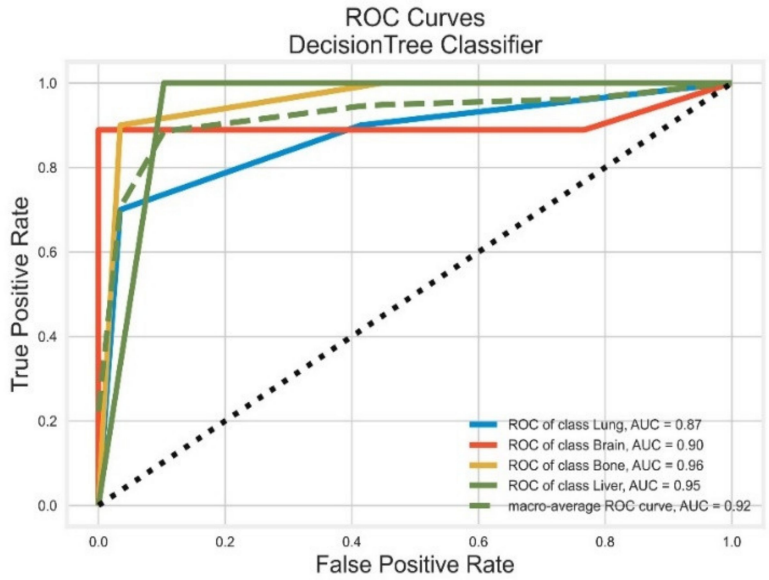

Figure 16. ROC Curves (Decision Tree Classifier).

Higher TPR at low FPR indicates that it is a good model, whereas area under curve (AUC) is the separability of a classifier. The greater the AUC, the better the model is. AUC = 0.92, indicating an excellent overall classifier.

3.2. Random Forest Classifier

Random forest classifier had an overall training accuracy of 98% and validation /test accuracy of 90%. The confusion matrix of the classifier is shown in Figure 17. RF Classifier has misclassified only four samples out of a total of 39. One of the lung samples is classified as liver and two of the bone samples are classified as brain and liver, respectively. At the same time, one liver sample is classified as bone.

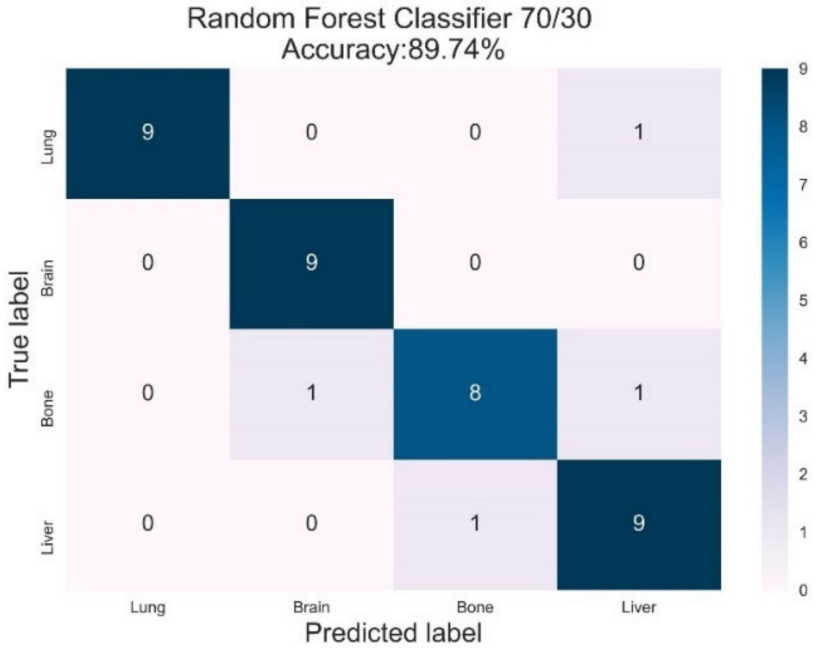

Figure 17. Confusion Matrix (Random Forest Classifier).

The classification report for RF classifier is shown in Figure 18. Class liver has a precision of 0.81 and bone with recall has a precision 0.80, thus reducing the overall F1 score for the liver and bone class to 0.85 and 0.84, respectively.

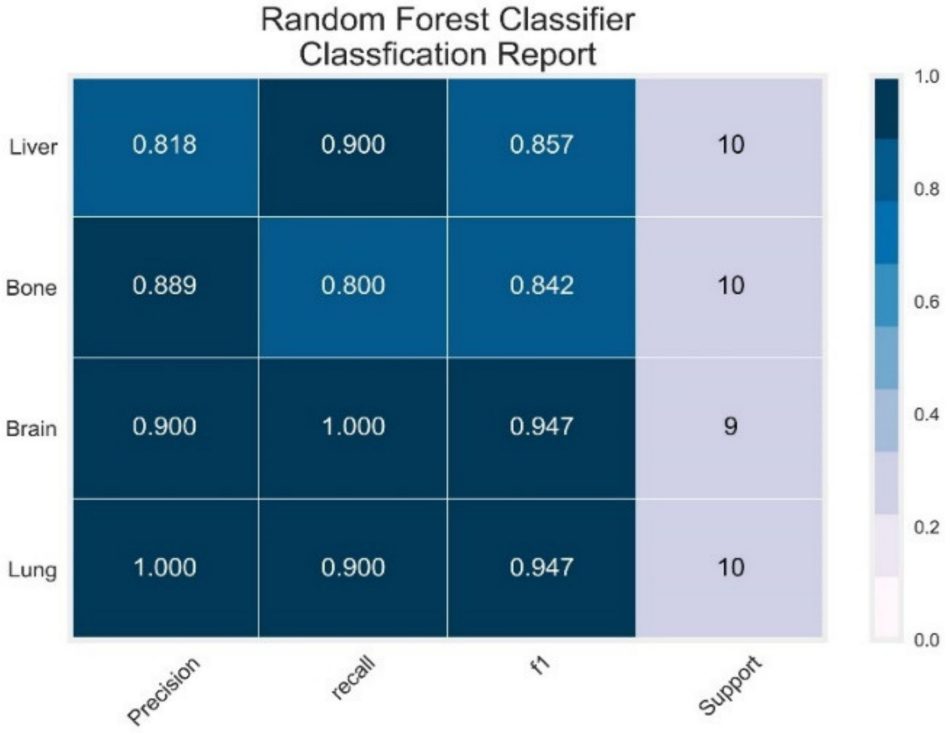

Figure 18. Classification Report (Random Forest Classifier).

The precision-recall curve for the RF classifier is shown below in Figure 19.

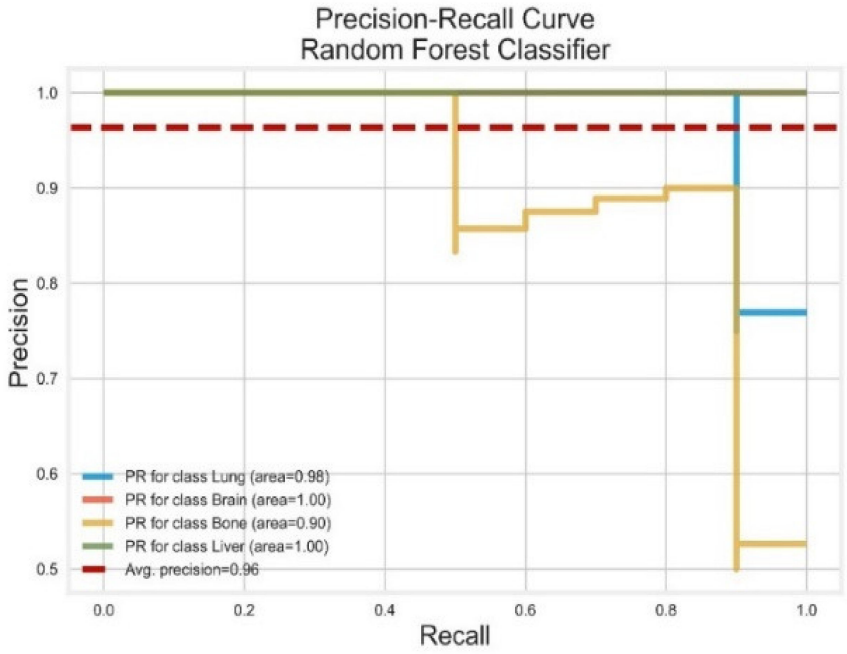

Figure 19. Precision-Recall Curve (Random Forest Classifier).

The RF classifier had an average precision of 0.96, whereas all classes have PR–AUC $\geq$ 0.90, exhibiting a good classifier model. The Receiver Operator Characteristic (ROC) is shown for RF Classifier in Figure 20.

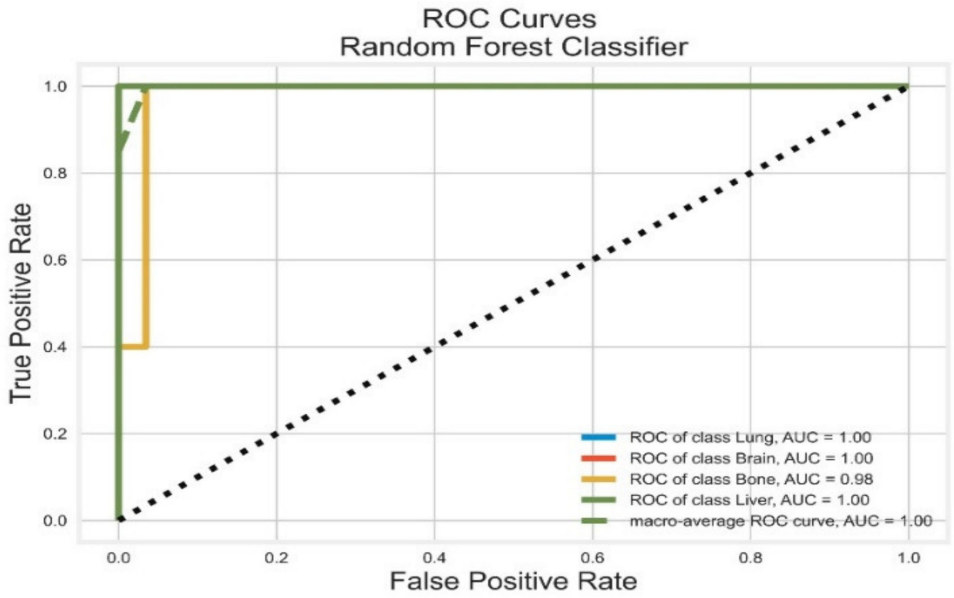

Figure 20. ROC Curves (Random Forest Classifier).

The average AUC = 1.0 for the RF classifier model where each class has an AUC $\geq$ 0.98. The classifier is showing better separability among all classes and prediction power.

3.3. K-Nearest Neighbour Classifier

K-nearest neighbor classifier reported a training accuracy of 92% and validation or test accuracy of 87%. The confusion matrix depicted in Figure 21 indicates the misclassified samples. Out of 39 samples, five were misclassified with two samples from the lung class misclassified as liver and three samples from the bone class misclassified with one sample in each lung, brain, and liver class.

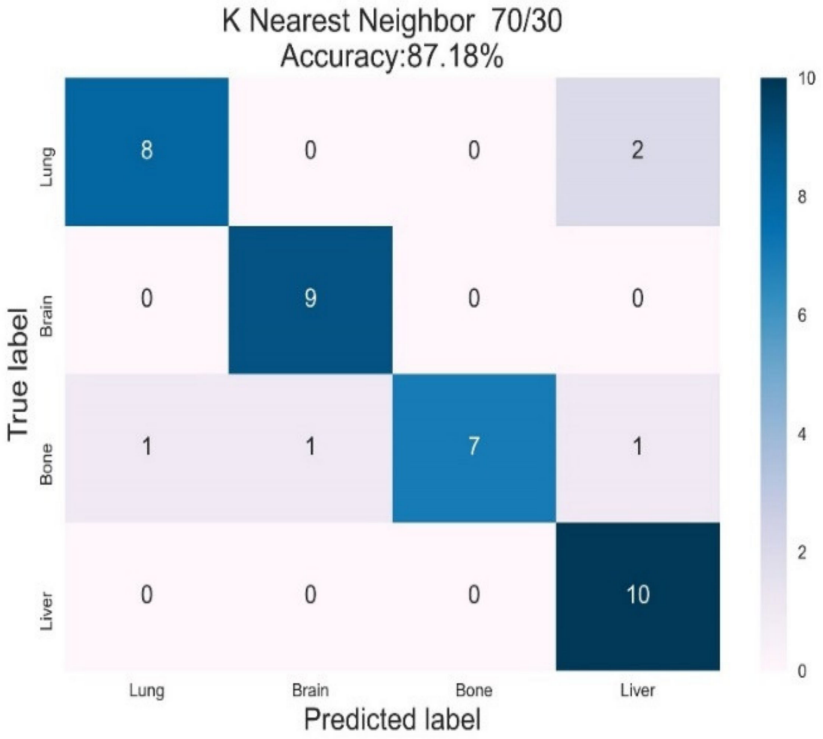

Figure 21. Confusion Matrix (K-Nearest Neighbor Classifier).

The classification report for the KNN classifier depicted in Figure 22 reveals the classifier's overall performance. Class liver has low precision of 0.77 with low recall parameters of 0.7 and 0.8 for bone and lung classes, thus affecting the overall F1 score of these classes.

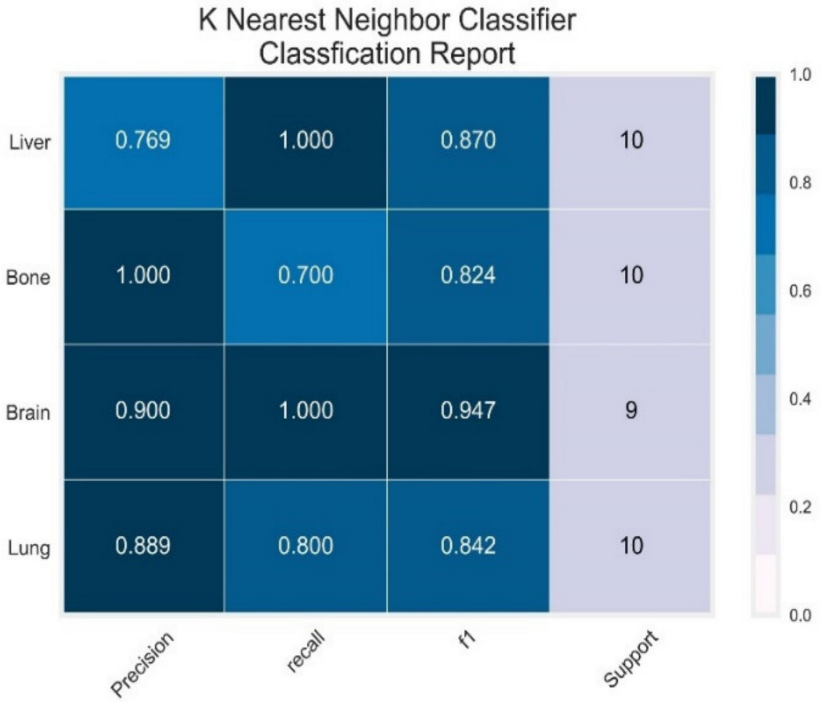

Figure 22. Classification Report (K-Nearest Neighbor Classifier).

The precision-recall curve for the KNN classifier is calculated using one-vs-rest method as shown in Figure 23.

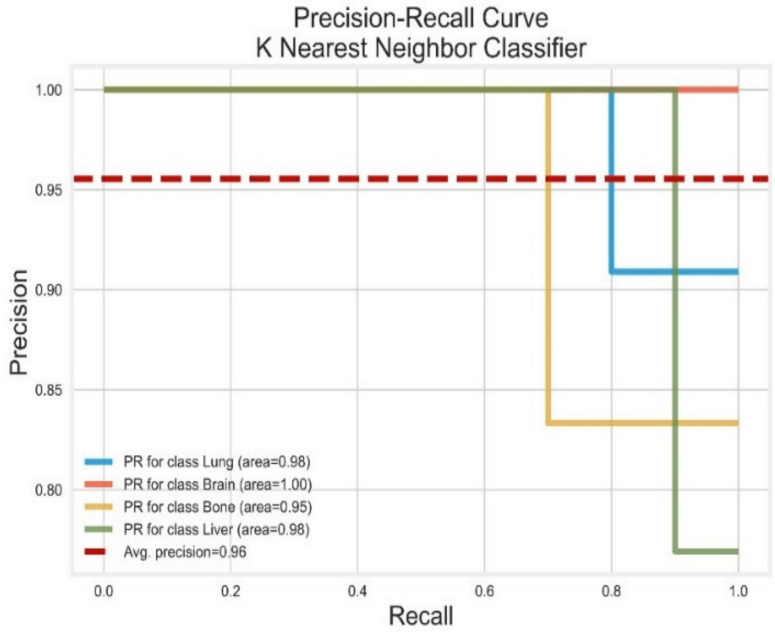

Figure 23. Precision-Recall Curve (K Nearest Neighbor Classifier).

The KNN classifier had an average precision of 0.96, whereas all classes have PR–AUC $\geq$ 0.95, exhibiting a good classifier model. The Receiver Operator Characteristic (ROC) is shown below for the KNN classifier in Figure 24 using the one-vs-all method.

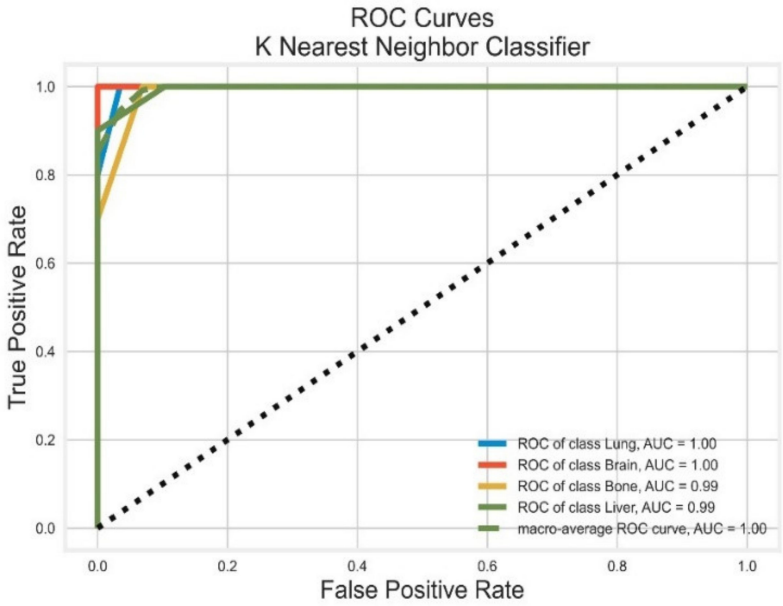

Figure 24. ROC Curves (K-Nearest Neighbor Classifier.

The average AUC = 1.0 for the KNN classifier model where all classes have an AUC $\geq$ 0.99, the greater the value of AUC better is the separability among all classes.

3.4. Support Vector Machines

The support vector machines classifier has shown excellent results with training accuracy of 100% and validation or test accuracy of 97%. The confusion matrix is depicted in Figure 25. Only one sample out of 39 samples is misclassified with one sample from the liver class reported under bone class.

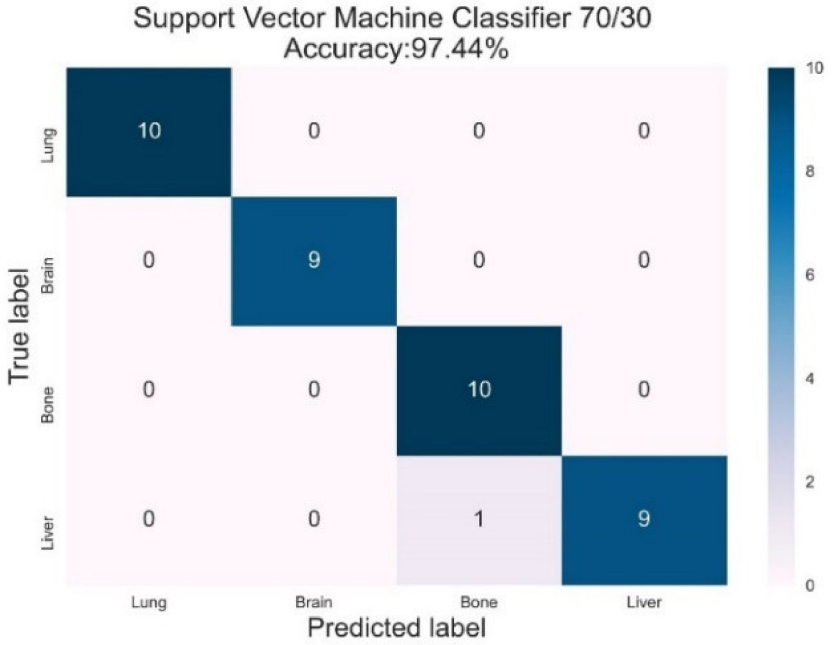

Figure 25. Confusion Matrix (Support Vector Machines Classifier).

The classification report for the SVM classifier visualized in Figure 26 revealed the excellent performance of the classifier. Precision and recall for all classes are $\geq 0.99 \rightarrow \leq 1.0$ and thus has a very high F1 score for each class.

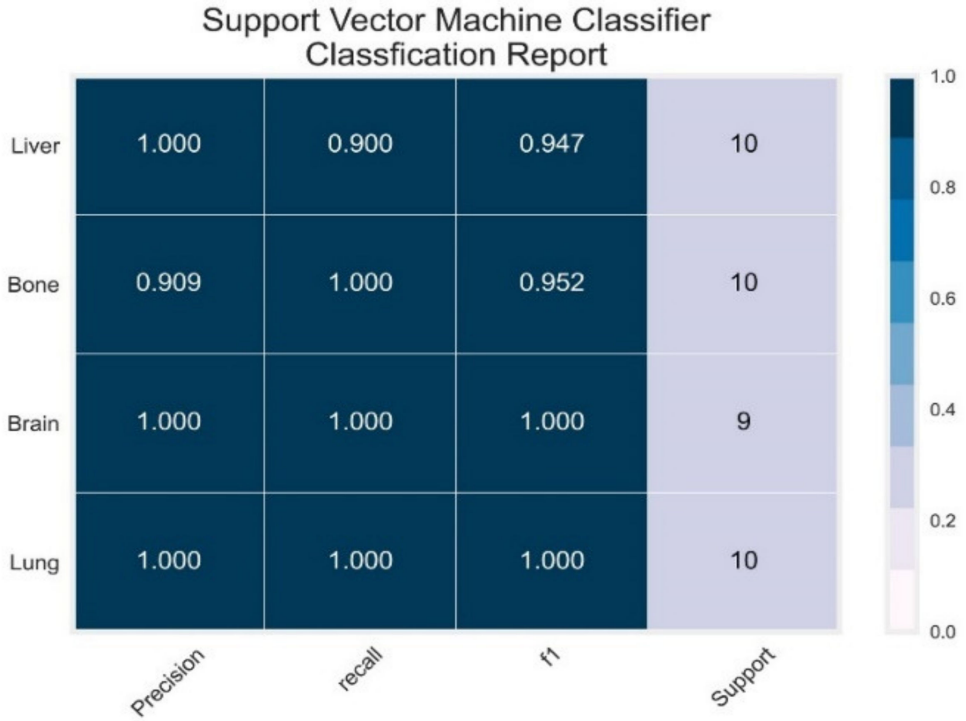

Figure 26. Classification Report (Support Vector Machines).

The precision-recall curve for the SVM classifier applying the one-vs-rest method is shown in Figure 27.

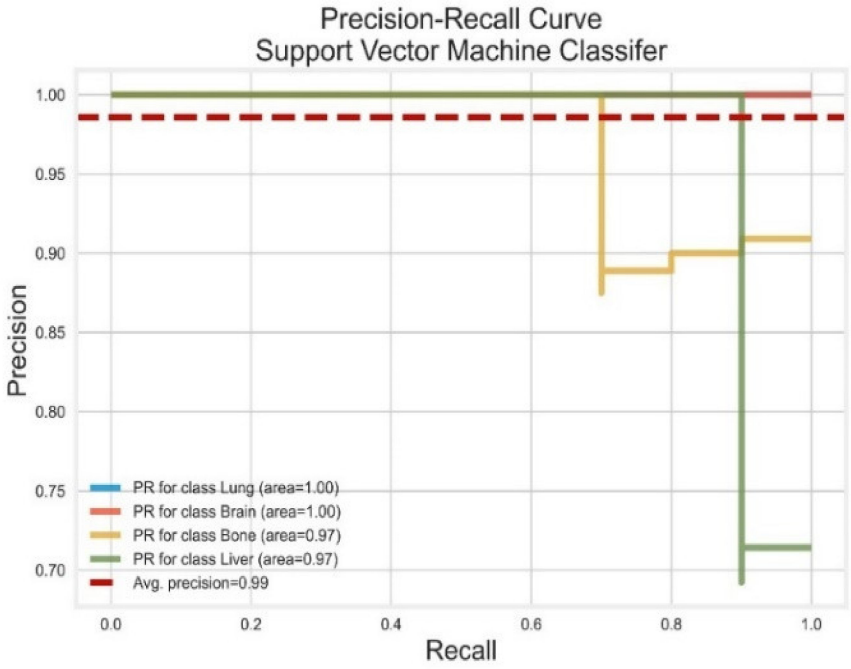

Figure 27. Precision-Recall Curve (Support Vector Machines).

The SVM classifier has an excellent average precision of 0.99, whereas all classes have PR–AUC $\geq$ 0.97, exhibiting an outstanding classifier model. The Receiver Operator Characteristic (ROC) is shown below for SVM Classifier in Figure 28 using the one-vs-all method.

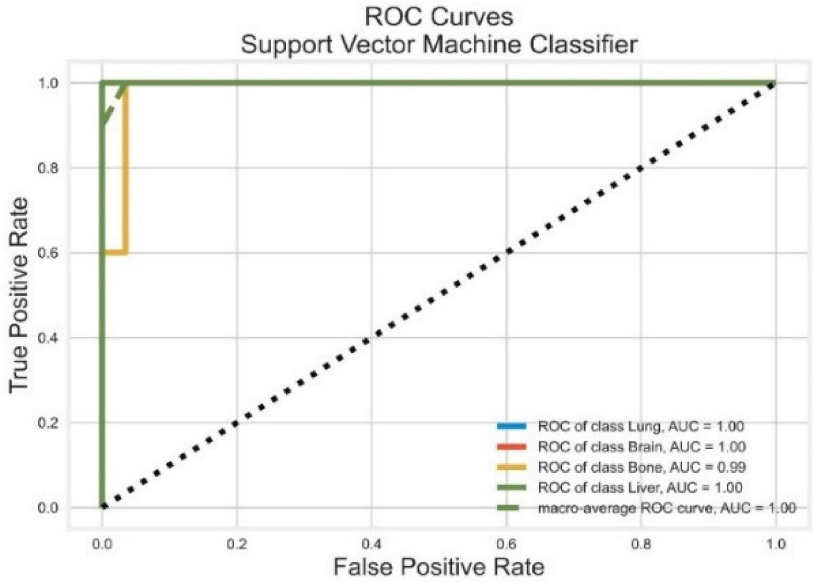

Figure 28. ROC Curves (Support Vector Machines).

The average for the SVM classifier model is AUC = 1.0 where the bone class has an AUC = 0.99, while all other classes have an AUC = 1.0. SVM is the best classifier with maximum separability among all classes. A tabular comparison is shown in Table 3. All four different classifiers have been compared based on precision, recall, F1 score, PR–AUC, ROC–AUC, and the number of misclassified samples. The SVM Classifier has outperformed all other classifiers as it has high accuracy, low variance, higher precision, recall, and F1 score. Moreover, this classifier has the least misclassified samples and the highest PR–AUC ROC–AUC values per class. All the AUC presented for different classifiers have been calculated with 95% confidence interval and $p > 0.05$.

Table 3. Evaluation Metrics for Comparative Analysis of Proposed Methods.

Classifier	Class	Precision	Recall	F1 Score	PR-AUC	ROC-AUC
DT	Lung	0.88	0.70	0.78	0.76	0.87
	Brain	1.00	0.89	0.94	1.00	0.90
	Bone	0.90	0.90	0.90	0.54	0.96
	Liver	0.77	1.00	0.87	0.70	0.95
					AVG Precision = 0.77	AVG ROC-AUC = 0.92
RF	Lung	1.00	0.90	0.95	0.98	1.00
	Brain	0.90	0.80	0.95	1.00	1.00
	Bone	0.89	1.00	0.84	0.90	0.98
	Liver	0.82	0.90	0.86	1.00	1.00
					AVG Precision = 0.96	AVG ROC-AUC = 1.00
KNN	Lung	0.89	0.80	0.84	0.98	1.00
	Brain	0.90	1.00	0.95	1.00	1.00
	Bone	1.00	0.70	0.82	0.95	0.99
	Liver	0.76	1.00	0.87	0.98	0.99
					AVG Precision = 0.96	AVG ROC-AUC = 1.00
SVM	Lung	1.00	1.00	1.00	1.00	1.00
	Brain	1.00	1.00	1.00	1.00	1.00
	Bone	0.91	1.00	0.95	0.97	0.99
	Liver	1.00	0.99	0.99	0.97	1.00
					AVG Precision = 0.99	AVG ROC-AUC = 1.00

Nevertheless, significant results have been achieved. Further research is required to validate these models on diverse datasets. However, computation power is one of the most significant constraints in handling gene expression microarray datasets.

4. Conclusions

This paper concluded that breast cancer prognosis often metastasizes towards bones, liver, brain, and lungs; a leading cause of death in women. It uniquely integrated machine learning and microarrays for the identification of breast cancer using K-nearest neighbors, missing values are imputed, recursive feature elimination with cross-validation, and class imbalance is handled by employing K-means SMOTE. This work successfully identified the 16 most essential Entrez Gene IDs responsible for predicting metastatic locations in the bones, brain, liver, and lungs. Extensive experiments were conducted on NCBI Gene Expression Omnibus GSE14020 and GSE54323 datasets. Multiple classification models were considered, and results were presented by considering reliable matrices such as ROC–AUC and PR–AUC, and F1-score. In the future, the authors aim to extend this work using more advanced learning approaches on multiple large datasets to identify the different metastasis stages.

Author Contributions: Conceptualization, F.R. and F.A., methodology, I.U.D., A.A. and S.U.D.; software, F.R.; validation, F.A.; formal analysis, I.U.D. and B.-S.K.; investigation, A.A. and B.-S.K.; resources, B.-S.K.; data curation, F.R. and F.A.; writing—original draft preparation, F.R. and F.A., writing— review and editing, I.U.D. and A.A.; visualization, B.-S.K. and A.A.; supervision, S.U.D.; project administration, S.U.D.; funding acquisition, B.-S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Research Foundation (NRF), Korea, under project BK21 FOUR (F21YY8102068); and in part by King Saud University, Riyadh, Saudi Arabia, through Researchers Supporting Project Number RSP-2022/184.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

AUC	Area under the curve
BC	Breast Cancer
BCBoM	Breast Cancer Bone Metastasis
BCBrM	Breast Cancer Brain Metastasis
BCLiM	Breast Cancer Liver Metastasis
BCLuM	Breast Cancer Lung Metastasis
CTCs	Circulating Tumor cells
CV	Cross validation
DT	Decision Tree supervised learning algorithm
FN	False Negative
FP	False Positive
GEO	Gene expression omnibus
K-Means	Unsupervised learning algorithm that partition dataset into K number of clusters
KNN	K-Nearest Neighbors supervised learning algorithm
NCBI	National Center for Biotechnology Information
NRF	National Research Foundation
PR	Precision Recall
RF	Random Forrest supervised learning algorithm
RFECV	Recursive Feature Elimination with cross validation
ROC	Receiver operating characteristic curve
SMOTE	Synthetic object oversampling technique
SVM	Support Vector Machine supervised learning algorithm
TN	True Negative
TP	True Positive

Appendix A. GSE14020

ENTREZ_GENE_ID	Index	1	10	100	1000	10,000	100,009,676	...	9990	9991	9992	9993	9994	9997	Metastasis
0	GSM352095	8.2339	6.0683	7.8848	6.8721	6.6923	5.1057	...	5.2606	7.9440	6.4849	9.3538	6.5735	10.7023	Lung
1	GSM352097	6.9053	6.8495	7.0798	9.1682	6.9491	5.3145	...	6.3746	7.6820	6.6459	8.7531	7.0091	8.8169	Brain
2	GSM352098	7.7113	7.2329	6.5495	7.2243	6.2732	5.5823	...	5.9297	8.2616	6.5661	9.7786	7.6423	10.8695	Brain
3	GSM352100	7.3770	6.7069	7.8392	7.3133	7.1324	5.1022	...	5.8855	7.6461	6.5335	8.9864	6.8501	9.6639	Bone
4	GSM352101	7.4527	6.8992	6.4535	10.0303	6.4948	5.1645	...	5.7241	7.4800	6.7075	9.0687	7.0794	9.8207	Brain
5	GSM352103	7.4528	6.8274	6.3866	10.8126	7.1971	5.4020	...	5.8136	7.3723	6.8072	8.3742	7.3350	9.8820	Bone
6	GSM352105	7.8187	6.6937	9.4577	10.7817	6.6365	5.1032	...	6.1665	7.1127	6.6030	8.4056	7.4552	10.1645	Bone
7	GSM352107	6.9245	6.4157	7.1858	7.8156	8.3712	5.0686	...	5.8752	7.5597	6.3730	8.8120	8.5782	10.5591	Brain
8	GSM352109	7.3767	6.4297	7.1149	7.5414	6.6557	5.1151	...	5.7456	7.4064	6.3695	8.5874	6.9917	9.9465	Bone
9	GSM352110	6.9355	6.8455	7.2018	8.0589	6.5079	5.2703	...	5.8681	9.3738	6.6825	9.0142	7.1229	9.2914	Brain
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
55	GSM352159	NaN	7.1089	6.6904	6.7784	5.1182	NaN	...	5.9607	7.0477	6.8578	7.4886	6.9233	8.5203	Bone
56	GSM352160	NaN	6.5573	7.5134	6.8305	5.4619	NaN	...	6.1137	6.7245	6.8011	7.8224	6.6020	9.0305	Lung
57	GSM352161	NaN	7.1461	6.9137	6.6266	5.0809	NaN	...	5.8923	6.5635	7.1701	8.3298	7.2257	9.5441	Lung
58	GSM352162	NaN	7.6670	6.4741	7.0452	5.1741	NaN	...	5.7948	6.6691	6.7381	7.5607	7.8597	7.8423	Liver
59	GSM352163	NaN	6.9949	7.4864	6.6844	5.5261	NaN	...	6.2890	6.7573	7.0070	7.5590	6.2727	9.0151	Bone
60	GSM352164	NaN	6.5512	7.3306	7.4419	5.4275	NaN	...	5.8670	6.2873	6.7903	8.0515	7.1116	8.6275	Lung
61	GSM352165	NaN	6.5440	7.3335	6.6873	5.9610	NaN	...	5.6563	6.9762	6.7983	8.1609	6.1113	9.2522	Lung
62	GSM352166	NaN	6.9497	6.6913	6.6506	6.2395	NaN	...	5.9061	6.2398	7.0367	7.8028	6.8671	8.4663	Lung
63	GSM352167	NaN	6.7676	7.1345	7.6319	6.5522	NaN	...	5.9233	6.5831	6.9459	8.0467	6.3893	8.5806	Bone
64	GSM352168	NaN	7.4883	6.9053	6.8512	5.6716	NaN	...	6.5621	6.0487	7.0821	7.7147	6.9370	9.2354	Lung

65 rows × 20,488 columns.

Appendix B. GSE54323

ENTREZ_GENE_ID	Index	1	10	100	1000	10,000	100,009,676	...	9990	9991	9992	9993	9994	9997	Metastasis
0	GSM1312928	7.5602	6.4469	7.3371	5.7804	4.4252	4.9310	...	6.4865	8.2751	5.9836	8.2346	6.8023	9.9452	Liver
1	GSM1312929	10.3885	8.2872	7.8423	7.3761	4.6025	5.1180	...	6.0557	6.9495	6.7754	8.0115	5.8184	9.1237	Liver
2	GSM1312930	6.7133	5.8366	5.7955	6.9065	4.6457	4.8781	...	5.4504	8.9469	6.3186	8.3371	6.3372	9.3204	Site
3	GSM1312931	6.2121	6.3439	5.7062	5.9918	4.3514	4.8626	...	6.3488	9.0194	6.4219	8.1964	6.2462	10.2259	Site
4	GSM1312932	5.8098	4.9986	6.9900	5.3402	4.6099	4.9493	...	5.2374	7.8907	6.1864	8.0367	6.0154	10.5537	Bone
5	GSM1312933	5.6922	5.0370	6.0000	5.3512	4.2550	4.9094	...	6.7374	8.4135	6.1085	8.2670	5.8228	9.6436	Bone
6	GSM1312934	6.2121	5.4978	7.2723	5.3844	4.3429	5.7149	...	5.9428	9.2940	6.2257	8.3165	6.6325	9.7894	Node
7	GSM1312935	6.4032	4.8821	7.3576	5.5975	5.3479	4.9806	...	6.1142	7.3487	6.1657	8.0086	6.4750	9.3352	Node
8	GSM1312936	6.5354	5.3353	6.2503	5.4279	4.3448	5.2540	...	5.5127	8.1144	6.2992	8.4049	6.3864	9.6644	Node
9	GSM1312937	6.1965	5.0868	7.0903	5.2549	4.9637	4.9261	...	6.8642	7.6783	7.0730	8.4742	5.4469	10.6389	Node
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
19	GSM1312947	6.0102	4.8673	7.4504	5.0322	5.0102	4.9572	...	8.3811	8.8105	6.2732	8.1065	6.2530	10.7216	Bone
20	GSM1312948	6.0744	5.0904	6.7679	5.7245	4.4632	4.7979	...	6.8297	8.1070	6.5280	8.1100	5.9623	8.9717	Bone
21	GSM1312949	6.7086	4.8407	6.6089	5.5950	4.3482	5.0114	...	6.9696	7.1899	6.3186	8.2858	5.6671	8.6960	Bone
22	GSM1312950	5.7694	5.5062	7.2207	5.2119	5.0936	5.3236	...	5.5929	8.3197	6.0860	8.1592	6.9327	9.4827	Node
23	GSM1312951	6.1343	5.4438	6.6683	5.5099	6.1355	5.5816	...	5.9490	8.3783	6.4996	8.3447	6.4786	9.3974	Node
24	GSM1312952	5.8563	4.8592	5.6757	5.3235	4.4993	5.2810	...	5.6794	8.8107	6.2094	8.1863	5.8603	8.3529	Liver
25	GSM1312953	8.8499	6.0498	5.9915	6.1030	4.3604	5.0026	...	5.3545	8.1970	6.2779	8.2310	5.9560	9.5760	Bone
26	GSM1312954	6.3137	4.7653	5.1842	5.2177	4.2325	5.7308	...	5.2766	9.3691	6.1371	8.0164	6.1084	8.2146	Bone
27	GSM1312955	5.9951	5.0517	6.4163	5.2682	4.3745	4.8983	...	6.2287	8.7048	6.0669	8.4942	5.6814	9.5736	Bone
28	GSM1312956	6.0545	4.7172	6.9508	5.1912	4.3765	4.8779	...	7.6965	8.8376	6.4431	8.2028	6.5684	9.2403	Bone

29 rows × 20,488 columns.

Appendix C. Merged and Normalized Dataset

	1	**10**	**100**	**1000**	**10000**	**1×10^8**	**10001**	**...**	**999**	**9990**	**9991**	**9992**	**9993**	**9994**	**9997**
0	1.1073	−0.5584	1.4254	−0.1679	0.968	−0.7103	0.3433	...	0.235	−1.4055	0.6487	−0.7894	1.7514	−0.47	1.4987
1	−0.4006	0.3714	0.2092	1.7604	1.2383	0.2766	0.7643	...	0.5823	0.5341	0.3564	−0.2172	0.7292	0.1851	−0.5383
2	0.5142	0.8277	−0.5921	0.1279	0.527	1.5429	0.784	...	1.3263	−0.2405	1.0032	−0.5009	2.4741	1.1371	1.6794
3	0.1348	0.2017	1.3565	0.2027	1.4312	−0.7269	1.0486	...	0.8783	−0.3174	0.3164	−0.6167	1.1261	−0.0541	0.3768
4	0.2206	0.4306	−0.7371	2.4844	0.7602	−0.4321	1.08	...	0.8878	−0.5984	0.131	0.002	1.2663	0.2907	0.5462
5	0.2208	0.3451	−0.8381	3.1413	1.4992	0.6906	1.4515	...	−3.5736	−0.4426	0.0108	0.3564	0.0845	0.6751	0.6125
6	0.6361	0.186	3.8018	3.1154	0.9093	−0.7219	2.3365	...	−3.1912	0.1717	−0.2788	−0.3698	0.138	0.8558	0.9176
7	−0.3789	−0.145	0.3693	0.6245	2.7348	−0.8856	1.0825	...	1.0557	−0.3354	0.2199	−1.1873	0.8295	2.5444	1.3439
8	0.1344	−0.1282	0.2622	0.3942	0.9295	−0.666	1.3149	...	0.9476	−0.561	0.0489	−1.1998	0.4473	0.1588	0.6821
9	−0.3664	0.3666	0.3935	0.8288	0.774	0.0678	0.6543	...	2.0178	−0.3477	2.2444	−0.087	1.1735	0.3561	−0.0257
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
76	−1.6765	−2.1117	0.5075	−1.1132	−0.8302	−2.4271	−1.1618	...	−0.8085	2.8795	−0.3754	−1.3423	−0.3784	−1.3842	1.1868
77	−1.7519	−1.7871	−0.2358	−1.4794	−0.1956	−0.0349	−1.4196	...	−0.6919	2.6152	1.5135	−1.5502	−0.057	−0.3818	0.9789
78	−1.4165	−1.9877	0.7691	−1.713	−0.8021	−1.4121	−0.6681	...	−1.1026	4.0275	1.6157	−1.5423	−0.371	−0.952	1.5195
79	−1.3437	−1.7222	−0.2621	−1.1316	−1.3777	−2.1654	−1.7002	...	−0.9386	1.3265	0.8307	−0.6363	−0.365	−1.389	−0.3711
80	−0.6238	−2.0195	−0.5024	−1.2403	−1.4987	−1.1558	−1.9665	...	−0.7943	1.5701	−0.1927	−1.3807	−0.0658	−1.8329	−0.6689
81	−1.5912	−1.9974	−1.9121	−1.4683	−1.3397	0.1184	−1.0834	...	0.0965	−0.6762	1.616	−1.7688	−0.2352	−1.5423	−1.0397
82	1.8065	−0.5804	−1.4351	−0.8137	−1.4859	−1.1977	−1.0668	...	0.8924	−1.2419	0.9312	−1.5253	−0.1591	−1.3984	0.2818
83	−1.0721	−2.1091	−2.6548	−1.5571	−1.6205	2.2447	−0.0992	...	0.7633	−1.3775	2.2391	−2.0259	−0.5243	−1.1693	−1.1891
84	−1.4337	−1.7683	−0.7933	−1.5147	−1.4711	−1.6908	−1.2391	...	0.1758	0.2801	1.4978	−2.2756	0.2887	−1.8114	0.2792
85	−1.3663	−2.1664	0.0143	−1.5794	−1.4689	−1.7872	−1.4259	...	−1.3731	2.8356	1.646	−0.938	−0.207	−0.4777	−0.0809

86 rows × 20,486 columns.

Appendix D. Dataset after Removing Correlated Features

	1	10	100	1000	10,000	1×10^8	10,001	...	9961	9962	997	9973	9984	999	9997
0	1.1073	−0.5584	1.4254	−0.1679	0.968	−0.7103	0.3433	...	0.2706	0.4551	1.6566	1.1436	−0.4051	0.235	1.4987
1	−0.4006	0.3714	0.2092	1.7604	1.2383	0.2766	0.7643	...	0.4908	0.4133	1.3221	0.6164	−0.2804	0.5823	−0.5383
2	0.5142	0.8277	−0.5921	0.1279	0.527	1.5429	0.784	...	1.1641	0.4141	0.6974	0.9285	0.2504	1.3263	1.6794
3	0.1348	0.2017	1.3565	0.2027	1.4312	−0.7269	1.0486	...	1.7457	−0.3479	0.0111	0.5555	−1.3193	0.8783	0.3768
4	0.2206	0.4306	−0.7371	2.4844	0.7602	−0.4321	1.08	...	1.4034	−0.6757	0.8695	0.2926	3.6107	0.8878	0.5462
5	0.2208	0.3451	−0.8381	3.1413	1.4992	0.6906	1.4515	...	0.1724	0.5311	−0.276	0.4835	0.33	−3.5736	0.6125
6	0.6361	0.186	3.8018	3.1154	0.9093	−0.7219	2.3365	...	0.3219	0.6182	−0.4591	0.0423	−0.6683	−3.1912	0.9176
7	−0.3789	−0.145	0.3693	0.6245	2.7348	−0.8856	1.0825	...	0.0195	−0.1824	−0.3309	2.046	−0.0084	1.0557	1.3439
8	0.1344	−0.1282	0.2622	0.3942	0.9295	−0.666	1.3149	...	0.9737	−0.5723	−0.6066	0.7488	−0.1313	0.9476	0.6821
9	−0.3664	0.3666	0.3935	0.8288	0.774	0.0678	0.6543	...	0.3707	0.221	0.2925	0.1104	−0.2957	2.0178	−0.0257
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
76	−1.6765	−2.1117	0.5075	−1.1132	−0.8302	−2.4271	−1.1618	...	−0.1819	−1.9179	2.44	0.735	−0.7166	−0.8085	1.1868
77	−1.7519	−1.7871	−0.2358	−1.4794	−0.1956	−0.0349	−1.4196	...	0.0729	−0.4649	−2.2167	−2.4931	1.0378	−0.6919	0.9789
78	−1.4165	−1.9877	0.7691	−1.713	−0.8021	−1.4121	−0.6681	...	0.9578	−0.3947	−1.3783	−1.6917	0.2525	−1.1026	1.5195
79	−1.3437	−1.7222	−0.2621	−1.1316	−1.3777	−2.1654	−1.7002	...	0.1721	−1.5257	1.3317	−1.0251	0.5284	−0.9386	−0.3711
80	−0.6238	−2.0195	−0.5024	−1.2403	−1.4987	−1.1558	−1.9665	...	0.5117	−1.6121	2.2701	−0.6082	0.2498	−0.7943	−0.6689
81	−1.5912	−1.9974	−1.9121	−1.4683	−1.3397	0.1184	−1.0834	...	−0.0482	−1.9324	−0.947	−1.6412	0.3456	0.0965	−1.0397
82	1.8065	−0.5804	−1.4351	−0.8137	−1.4859	−1.1977	−1.0668	...	0.2392	0.4666	−0.4142	2.4051	0.4809	0.8924	0.2818
83	−1.0721	−2.1091	−2.6548	−1.5571	−1.6205	2.2447	−0.0992	...	−0.4106	−1.118	−2.1279	−1.5875	0.4401	0.7633	−1.1891
84	−1.4337	−1.7683	−0.7933	−1.5147	−1.4711	−1.6908	−1.2391	...	−1.3745	−2.1706	−0.6876	−0.8633	1.3487	0.1758	0.2792
85	−1.3663	−2.1664	0.0143	−1.5794	−1.4689	−1.7872	−1.4259	...	0.2409	−2.3795	0.45	−0.2579	0.5958	−1.3731	−0.0809

86 rows × 6602 columns.

Appendix E. Reduced Dimensions after RFECV

	222,865	25,973	2670	347,733	4131	4283	4318	441,150	51,011	6439	729,887	81,035	8120	81,575	8817	90,865
0	0.198	0.9758	0.3665	−0.2696	1.0458	0.9783	−0.6444	3.782	3.171	0.3832	4.6323	−0.4311	−0.2163	−0.1137	−0.1295	0.7515
1	−0.0865	1.1	0.7864	3.312	1.2416	0.5305	0.9294	0.6143	0.9198	0.1937	−0.1079	−0.3811	4.3088	1.2101	−0.2767	−0.4483
2	1.9646	1.1233	1.0511	−0.0854	0.9364	−0.79	−1.2065	0.9356	0.1308	−0.0736	2.1	−0.71	0.2795	0.9134	−0.3304	−0.142
3	−1.192	−0.6367	0.2691	−0.4562	1.7566	−0.7846	1.4329	−0.7015	0.4727	−0.1349	−0.7771	1.5955	−0.3288	2.6731	3.8883	1.3303
4	0.1266	−0.0604	1.7196	0.0398	1.8735	−0.659	0.0221	0.1035	0.4401	−0.0272	2.0996	−0.2556	−0.4302	−0.0581	0.8987	−0.5004
5	−0.5906	0.3491	1.4248	0.2006	2.0667	−0.7052	−0.5249	−0.2215	−0.3393	0.1466	−0.8944	0.6907	−0.2445	−0.2771	−0.0903	−0.4262
6	−0.8871	0.1041	0.1751	−0.7043	1.6466	−0.3017	0.1538	−0.4897	0.4002	0.0265	−0.4333	0.4794	−0.0686	1.0238	−0.2101	−0.2477
7	0.4481	0.0621	0.9934	1.1176	1.0911	−0.7203	0.0915	1.7954	1.4472	−0.0197	2.5998	−0.3057	−0.5338	1.8907	0.3507	−0.4034
8	−0.2063	1.0118	0.0792	−0.7138	1.0369	1.4341	1.8265	−0.2083	0.2059	−0.0171	−0.8552	1.9019	−0.5076	0.847	0.2876	0.1463
9	−0.2731	1.155	2.0704	1.0202	2.5062	−0.6621	−0.5338	−0.1218	0.7685	0.1553	0.1663	−0.5509	−0.2045	1.5289	0.1918	−0.2485
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
76	−1.012	−1.9397	−1.1871	−0.9757	−1.8074	−0.5485	1.1886	−1.6693	−1.1848	−0.9253	−0.8543	−0.6911	−0.4686	−0.2061	−1.3553	−0.7851
77	−1.9576	−0.0641	−1.302	−0.7803	0.1777	0.6541	1.6583	−1.4042	−0.9199	−0.9623	−1.659	0.5504	−0.8571	−0.2125	−1.4715	−1.1276
78	−1.786	−1.2262	−1.3476	−0.3278	−0.4901	0.5951	1.933	−1.0345	−1.1996	−0.9378	−1.8962	1.3793	−0.8242	−0.7745	−1.7589	−0.9362
79	−0.2186	−1.1662	−1.15	−0.7646	−0.3131	−1.3009	1.7065	−2.237	−1.5943	−0.9207	−0.9506	0.7116	−0.9004	0.2218	−1.887	−1.096
80	1.0808	−0.597	−0.8584	−0.8226	−1.4641	−1.2374	0.7571	−1.4812	−2.0239	−0.7141	−1.4048	0.3666	−0.9214	−1.3547	0.2259	−1.0798
81	−1.3624	0.1524	−1.1858	−0.8277	−1.4117	−1.2232	−1.2171	−0.8709	−0.8319	−0.8716	−0.6282	1.7031	−1.7363	−0.7022	−1.9667	−1.0593
82	−1.2753	−0.6617	−1.2841	−0.6723	0.2354	0.0003	−1.3449	−0.3184	−0.8008	−0.8581	0.298	−1.1678	−0.6246	−0.3164	1.769	−0.4109
83	−1.0755	0.189	−1.3647	−0.902	−1.5986	−1.9597	−0.8797	−1.0385	−0.5315	−0.7335	−0.8141	1.275	−1.888	−0.2927	−1.5776	−1.1293
84	−1.8085	−1.0142	−1.2811	−0.9334	−1.0717	−0.4323	0.9235	−2.1046	−1.0034	−0.9462	−0.8141	−1.2629	−0.9565	−0.2507	−1.7703	−0.6834
85	−0.6364	−1.8746	−0.8828	−0.8022	−1.9914	0.1244	1.8855	−1.3999	−1.2799	−0.7505	−0.6868	−1.4617	−0.0416	−1.7382	−1.2005	−0.5755

86 rows × 16 columns.

Appendix F. Oversampled Dataset after Applying K-Mean Smote

	222,865	25,973	2670	347,733	4131	4283	4318	441,150	51,011	6439	729,887	81,035	8120	81,575	8817	90,865	Metastasis
0	0.198	0.9758	0.3665	−0.2696	1.0458	0.9783	−0.6444	3.782	3.171	0.3832	4.6323	−0.4311	−0.2163	−0.1137	−0.1295	0.7515	Lung
1	−0.0865	1.1	0.7864	3.312	1.2416	0.5305	0.9294	0.6143	0.9198	0.1937	−0.1079	−0.3811	4.3088	1.2101	−0.2767	−0.4483	Brain
2	1.9646	1.1233	1.0511	−0.0854	0.9364	−0.79	−1.2065	0.9356	0.1308	−0.0736	2.1	−0.71	0.2795	0.9134	−0.3304	−0.142	Brain
3	−1.192	−0.6367	0.2691	−0.4562	1.7566	−0.7846	1.4329	−0.7015	0.4727	−0.1349	−0.7771	1.5955	−0.3288	2.6731	3.8883	1.3303	Bone
4	0.1266	−0.0604	1.7196	0.0398	1.8735	−0.659	0.0221	0.1035	0.4401	−0.0272	2.0996	−0.2556	−0.4302	−0.0581	0.8987	−0.5004	Brain
5	−0.5906	0.3491	1.4248	0.2006	2.0667	−0.7052	−0.5249	−0.2215	−0.3393	0.1466	−0.8944	0.6907	−0.2445	−0.2771	−0.0903	−0.4262	Bone
6	−0.8871	0.1041	0.1751	−0.7043	1.6466	−0.3017	0.1538	−0.4897	0.4002	0.0265	−0.4333	0.4794	−0.0686	1.0238	−0.2101	−0.2477	Bone
7	0.4481	0.0621	0.9934	1.1176	1.0911	−0.7203	0.0915	1.7954	1.4472	−0.0197	2.5998	−0.3057	−0.5338	1.8907	0.3507	−0.4034	Brain
8	−0.2063	1.0118	0.0792	−0.7138	1.0369	1.4341	1.8265	−0.2083	0.2059	−0.0171	−0.8552	1.9019	−0.5076	0.847	0.2876	0.1463	Bone
9	−0.2731	1.155	2.0704	1.0202	2.5062	−0.6621	−0.5338	−0.1218	0.7685	0.1553	0.1663	−0.5509	−0.2045	1.5289	0.1918	−0.2485	Brain
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
119	−1.2187	−2.5334	−1.328	1.6502	−0.1868	2.3117	−0.7413	−1.161	−0.9107	−0.8275	−0.8525	−0.9858	−1.3184	−1.3744	−1.9571	0.8591	Liver
120	−0.9832	−1.4084	−1.022	0.3152	−1.2142	1.989	−0.7787	−1.2526	−0.7445	−0.5679	−0.9831	−1.2853	−1.8635	−1.9452	−1.5483	3.7285	Liver
121	−1.4467	−1.1812	−1.3396	0.4043	−0.9398	1.6933	−1.1969	−1.693	−0.9057	−0.8883	−1.0399	−1.4553	−1.4607	−2.0827	−2.2189	0.1661	Liver
122	−0.5328	−1.365	−0.9757	1.0245	−0.9261	2.2429	−0.9658	−1.1536	−0.8336	−0.5485	−0.9617	−1.2656	−1.912	−2.0633	−1.2153	3.7183	Liver
123	−0.7801	−1.3888	−1.0011	0.6352	−1.0842	2.1035	−0.8631	−1.2079	−0.7847	−0.5592	−0.9735	−1.2764	−1.8854	−1.9984	−1.398	3.7239	Liver
124	−1.1877	−0.5688	−1.1857	−0.0131	−1.3437	1.6019	−1.3146	−1.7328	−0.8642	−0.7686	−1.1032	−1.614	−1.7331	−2.4309	−1.9283	1.4033	Liver
125	−1.331	−2.2994	−1.2769	1.0944	−0.5203	2.1472	−0.6812	−1.2176	−0.8412	−0.7769	−0.8893	−1.0595	−1.4218	−1.458	−1.988	1.5002	Liver
126	−1.0673	−2.7155	−1.3628	2.193	0.1203	2.4782	−0.815	−1.1022	−0.9786	−0.8655	−0.8199	−0.9239	−1.2419	−1.3232	−1.8872	0.3424	Liver
127	−1.5902	−0.5177	−1.3223	−0.3542	−1.3972	1.3651	−1.3603	−1.9386	−0.8735	−0.8917	−1.1338	−1.6824	−1.5646	−2.4134	−2.3412	0.1555	Liver
128	−1.7172	−1.307	−1.126	−0.8419	−1.6835	1.5209	−0.5981	−1.5013	−0.6315	−0.638	−1.0401	−1.3814	−1.7628	−1.8607	−2.1368	3.2783	Liver

129 rows × 17 columns.

References

1. Medeiros, B.; Allan, A.L. Molecular Mechanisms of Breast Cancer Metastasis to the Lung: Clinical and Experimental Perspectives. *Int. J. Mol. Sci.* **2019**, *20*, 2272. [CrossRef] [PubMed]
2. Chaffer, C.L.; Weinberg, R.A. A Perspective on Cancer Cell Metastasis. *Science* **2011**, *331*, 1559–1564. [CrossRef] [PubMed]
3. JPMA—Journal of Pakistan Medical Association. Available online: https://jpma.org.pk/article-details/1863 (accessed on 13 February 2021).
4. Menhas, R.; Umer, S. Breast Cancer among Pakistani Women. *Iran. J. Public Health* **2015**, *44*, 586–587. [PubMed]
5. Zaheer, S.; Shah, N.; Maqbool, S.A.; Soomro, N.M. Estimates of Past and Future Time Trends in Age-Specific Breast Cancer Incidence among Women in Karachi, Pakistan: 2004–2025. *BMC Public Health* **2019**, *19*, 1001. [CrossRef]
6. Lambert, A.W.; Pattabiraman, D.R.; Weinberg, R.A. Emerging Biological Principles of Metastasis. *Cell* **2017**, *168*, 670–691. [CrossRef]
7. Hess, K.R.; Varadhachary, G.R.; Taylor, S.H.; Wei, W.; Raber, M.N.; Lenzi, R.; Abbruzzese, J.L. Metastatic Patterns in Adenocarcinoma. *Cancer* **2006**, *106*, 1624–1633. [CrossRef]
8. Wu, Q.; Li, J.; Zhu, S.; Wu, J.; Chen, C.; Liu, Q.; Wei, W.; Zhang, Y.; Sun, S. Breast Cancer Subtypes Predict the Preferential Site of Distant Metastases: A SEER Based Study. *Oncotarget* **2017**, *8*, 27990–27996. [CrossRef]
9. Schlappack, O.K.; Baur, M.; Steger, G.; Dittrich, C.; Moser, K. The Clinical Course of Lung Metastases from Breast Cancer. *Klin. Wochenschr.* **1988**, *66*, 790–795. [CrossRef]
10. Xiao, W.; Zheng, S.; Liu, P.; Zou, Y.; Xie, X.; Yu, P.; Tang, H.; Xie, X. Risk Factors and Survival Outcomes in Patients with Breast Cancer and Lung Metastasis: A Population-Based Study. *Cancer Med.* **2018**, *7*, 922–930. [CrossRef]
11. GSE14020—NCBI. Available online: https://www.ncbi.nlm.nih.gov/search/all/?term=GSE14020 (accessed on 6 December 2020).
12. GSE54323—NCBI. Available online: https://www.ncbi.nlm.nih.gov/search/all/?term=GSE54323 (accessed on 6 December 2020).
13. Daoud, M.; Mayo, M. A Survey of Neural Network-Based Cancer Prediction Models from Microarray Data. *Artif. Intell. Med.* **2019**, *97*, 204–214. [CrossRef]
14. Yazici, H.; Akin, B. Molecular Genetics of Metastatic Breast Cancer. In *Tumour Progression and Metastasis*; 2020; Available online: https://books.google.com.hk/books?hl=zh-CN&lr=&id=WXL8DwAAQBAJ&oi=fnd&pg=PA33&dq=Molecular+Genetics+of+Metastatic+Breast+Cancer.+In+Tumou&ots=fD07Myo0Zn&sig=N7UQpRfEosuIQxpTXI4KZx755Yc&redir_esc=y&hl=zh-CN&sourceid=cndr#v=onepage&q=Molecular%20Genetics%20of%20Metastatic%20Breast%20Cancer.%20In%20Tumou&f=false (accessed on 10 June 2022).
15. Saunus, J.M.; Momeny, M.; Simpson, P.T.; Lakhani, S.R.; Da Silva, L. Molecular Aspects of Breast Cancer Metastasis to the Brain. *Genet. Res. Int.* **2011**, *2011*, 1–9. [CrossRef]
16. Jin, X.; Mu, P. Targeting Breast Cancer Metastasis. *Breast Cancer Basic Clin. Res.* **2015**, *9*, 23–34. [CrossRef]
17. Macedo, F.; Ladeira, K.; Pinho, F.; Saraiva, N.; Bonito, N.; Pinto, L.; Gonçalves, F. Bone Metastases: An Overview. *Oncol. Rev.* **2017**, *11*, 321. [CrossRef]
18. Ma, R.; Feng, Y.; Lin, S.; Chen, J.; Lin, H.; Liang, X.; Zheng, H.; Cai, X. Mechanisms Involved in Breast Cancer Liver Metastasis. *J. Transl. Med.* **2015**, *13*, 64. [CrossRef]
19. Zhao, H.Y.; Gong, Y.; Ye, F.G.; Ling, H.; Hu, X. Incidence and Prognostic Factors of Patients with Synchronous Liver Metastases upon Initial Diagnosis of Breast Cancer: A Population-Based Study. *Cancer Manag. Res.* **2018**, *10*, 5937–5950. [CrossRef]
20. Pedrosa, R.M.S.M.; Mustafa, D.A.; Soffietti, R.; Kros, J.M. Breast Cancer Brain Metastasis: Molecular Mechanisms and Directions for Treatment. *Neuro. Oncol.* **2018**, *20*, 1439–1449. [CrossRef]
21. Brosnan, E.M.; Anders, C.K. Understanding Patterns of Brain Metastasis in Breast Cancer and Designing Rational Therapeutic Strategies. *Ann. Transl. Med.* **2018**, *6*, 163. [CrossRef]
22. Stella, G.M.; Kolling, S.; Benvenuti, S.; Bortolotto, C. Lung-Seeking Metastases. *Cancers* **2019**, *11*, 1010. [CrossRef]
23. Jin, L.; Han, B.; Siegel, E.; Cui, Y.; Giuliano, A.; Cui, X. Breast Cancer Lung Metastasis: Molecular Biology and Therapeutic Implications. *Cancer Biol. Ther.* **2018**, *19*, 858–868. [CrossRef]
24. Edgar, R.; Domrachev, M.; Lash, A.E. Gene Expression Omnibus: NCBI Gene Expression and Hybridization Array Data Repository. *Nucleic Acids Res.* **2002**, *30*, 207–210. [CrossRef]
25. *Affimetrix Human Genome U133 Arrays the Most Comprehensive Coverage of the Human Genome in Two Flexible Formats: Single-Array Cartridges and Multi-Array Plates*; 2017; Available online: https://www.thermofisher.com/ (accessed on 1 January 2022).
26. Maglott, D.; Ostell, J.; Pruitt, K.D.; Tatusova, T. Entrez Gene: Gene-Centered Information at NCBI. *Nucleic Acids Res.* **2011**, *39*. [CrossRef]
27. SOFT—GEO—NCBI. Available online: https://www.ncbi.nlm.nih.gov/geo/info/soft.html (accessed on 2 January 2021).
28. GEOparse—GEOparse 1.2.0 Documentation. Available online: https://geoparse.readthedocs.io/en/latest/introduction.html (accessed on 19 December 2020).
29. Liew, A.W.-C.; Law, N.-F.; Yan, H. Missing Value Imputation for Gene Expression Data: Computational Techniques to Recover Missing Data from Available Information. *Brief. Bioinform.* **2011**, *12*, 498–513. [CrossRef]
30. Bonaccorso, G. *Machine Learning Algorithms: A Reference Guide to Popular Algorithms for Data Science and Machine Learning*; Packt Publishing: Birmingham, UK, 2017; ISBN 1785889621.
31. Lin, W.C.; Tsai, C.F. Missing Value Imputation: A Review and Analysis of the Literature (2006–2017). *Artif. Intell. Rev.* **2020**, *53*, 1487–1509. [CrossRef]

32. Troyanskaya, O.; Cantor, M.; Sherlock, G.; Brown, P.; Hastie, T.; Tibshirani, R.; Botstein, D.; Altman, R.B. Missing Value Estimation Methods for DNA Microarrays. *Bioinformatics* **2001**, *17*, 520–525. [CrossRef] [PubMed]
33. Hastie, T.; Tibshirani, R.; Sherlock, G.; Eisen, M.; Brown, P.; Botstein, D. Imputing Missing Data for Gene Expression Arrays. *Stanford Univ. Stat. Dep. Tech. Rep.* **2006**, *3*, 27. Available online: http://www.stat.stanford.edu/Hast.pdf/cll/qxd. (accessed on 2 January 2021).
34. Sklearn.Impute.KNNImputer—Scikit-Learn 0.24.0 Documentation. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.impute.KNNImputer.html (accessed on 8 January 2021).
35. Dash, R.; Misra, B.B. Performance Analysis of Clustering Techniques over Microarray Data: A Case Study. *Phys. A Stat. Mech. Its Appl.* **2018**, *493*, 162–176. [CrossRef]
36. Mukaka, M.M. Statistics Corner: A Guide to Appropriate Use of Correlation Coefficient in Medical Research. *Malawi Med. J.* **2012**, *24*, 69–71. [PubMed]
37. Darst, B.F.; Malecki, K.C.; Engelman, C.D. Using Recursive Feature Elimination in Random Forest to Account for Correlated Variables in High Dimensional Data. *BMC Genet.* **2018**, *19*, 65. [CrossRef]
38. Li, Z.; Xie, W.; Liu, T. Efficient Feature Selection and Classification for Microarray Data. *PLoS ONE* **2018**, *13*, 1–21. [CrossRef]
39. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-Sampling Technique. *J. Artif. Int. Res.* **2002**, *16*, 321–357. [CrossRef]
40. Douzas, G.; Bacao, F.; Last, F. Improving Imbalanced Learning through a Heuristic Oversampling Method Based on K-Means and {SMOTE}. *Inf. Sci. (NY)* **2018**, *465*, 1–20. [CrossRef]
41. Riaz, F. Integration-of-Machine-Learning-and-Microarrays-for-the-Identification-of-Breast-Cancer-in-Vital-Org/Oversampled_afterKmeanSmote_data.csv at Master Faisalriazz/Integration-of-Machine-Learning-and-Microarrays-for-the-Identification-of-Breast-Cancer-in-Vi. Available online: https://github.com/faisalriazz/Integration-of-Machine-Learning-and-Microarrays-for-the-Identification-of-Breast-Cancer-in-Vital-Org/blob/master/oversampled_afterKmeanSmote_data.csv (accessed on 16 July 2021).
42. Ritu, A. Latiyan Shiwam Prediction of Breast Cancer Using Different Machine Learning Algorithms. In *Proceedings of 6th International Conference on Recent Trends in Computing*; Mahapatra, R.P., Panigrahi, B.K., Kaushik, B.K., Roy, S., Eds.; Springer Nature Pte. Ltd.: Berlin, Germany, 2021; pp. 369–383. ISBN 978-981-334-501-0.
43. Rajaguru, H.; Sannasi Chakravarthy, S.R. Analysis of Decision Tree and K-Nearest Neighbor Algorithm in the Classification of Breast Cancer. *Asian Pacific. J. Cancer Prev.* **2019**, *20*, 3777–3781. [CrossRef]
44. Al-Salihy, N.K.; Ibrikci, T. Classifying Breast Cancer by Using Decision Tree Algorithms. *ACM Int. Conf. Proc. Ser.* **2017**, 144–148. [CrossRef]
45. Nel, I.; Morawetz, E.W.; Tschodu, D.; Käs, J.A.; Aktas, B. The Mechanical Fingerprint of Circulating Tumour Cells (Ctcs) in Breast Cancer Patients. *Cancers* **2021**, *13*, 1119. [CrossRef]
46. Andreas, C.M.; Sarah, G. Introduction to Machine Learning with Python. In *Introduction to Machine Learning with Python*; O'Reilly Media, Inc.: Sevastopol, CA, USA, 2016; pp. 83–84. ISBN 9781449369415.
47. Huang, S.; Nianguang, C.A.I.; Penzuti Pacheco, P.; Narandes, S.; Wang, Y.; Wayne, X.U. Applications of Support Vector Machine (SVM) Learning in Cancer Genomics. *Cancer Genom. Proteom.* **2018**, *15*, 41–51. [CrossRef]