Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 2836236, 17 pages
https://doi.org/10.1155/2020/2836236

Research Article

The Effect of Training and Testing Process on Machine Learning in Biomedical Datasets

Muhammed Kürşad Uçar,[1] Majid Nour,[2] Hatem Sindi,[3] and Kemal Polat[4]

[1]*Electrical and Electronics Engineering, Faculty of Engineering, Sakarya University, Sakarya 54187, Turkey*
[2]*Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia*
[3]*King Abdulaziz University, Knowledge-Economy, and Technology Transfer Center, Jeddah 21589, Saudi Arabia*
[4]*Department of Electrical and Electronics Engineering, Faculty of Engineering, Bolu Abant Izzet Baysal University, Bolu 14280, Turkey*

Correspondence should be addressed to Kemal Polat; kpolat@ibu.edu.tr

Received 21 November 2019; Revised 2 March 2020; Accepted 28 April 2020; Published 13 May 2020

Academic Editor: Azeddine Beghdadi

Training and testing process for the classification of biomedical datasets in machine learning is very important. The researcher should choose carefully the methods that should be used at every step. However, there are very few studies on method choices. The studies in the literature are generally theoretical. Besides, there is no useful model for how to select samples in the training and testing process. Therefore, there is a need for resources in machine learning that discuss the training and testing process in detail and offer new recommendations. This article provides a detailed analysis of the training and testing process in machine learning. The article has the following sections. The third section describes how to prepare the datasets. Four balanced datasets were used for the application. The fourth section describes the rate and how to select samples at the training and testing stage. The fundamental sampling theorem is the subject of statistics. It shows how to select samples. In this article, it has been proposed to use sampling methods in machine learning training and testing process. The fourth section covers the theoretic expression of four different sampling theorems. Besides, the results section has the results of the performance of sampling theorems. The fifth section describes the methods by which training and pretest features can be selected. In the study, three different classifiers control the performance. The results section describes how the results should be analyzed. Additionally, this article proposes performance evaluation methods to evaluate its results. This article examines the effect of the training and testing process on performance in machine learning in detail and proposes the use of sampling theorems for the training and testing process. According to the results, datasets, feature selection algorithms, classifiers, training, and test ratio are the criteria that directly affect performance. However, the methods of selecting samples at the training and testing stages are vital for the system to work correctly. In order to design a stable system, it is recommended that samples should be selected with a stratified systematic sampling theorem.

1. Introduction

With the development of technology, the processes in every field have begun to become complicated. Collected data sizes began to increase. The data has grown so much that the cloud computing sector has emerged. With the growth of data, meaningful data has become more complicated. A field of data mining has emerged to make sense of the data. Data mining is a set of operations to derive meaning from data [1]. Data mining reveals the patterns within the data, the causes of the events, or the situations in which they relate [1]. For example, we can determine that supermarket customers prefer to buy products from middle shelves by data mining.

Machine learning is the generic name of the methods used to make sense of the data, to make decisions, and to predict [2]. How do we check if the machine has learned? Of course, we can learn by asking him a question. Training and questioning processes are generally called Training and Test

Processes [1]. The large size of the data has complicated the training and testing processes in machine learning. Every step needs a different process. Each process is involved in itself. There are hundreds of methods for each process. It is complicated to find and use the necessary method for each process. There are many sources of machine learning in the literature [1–4]. These resources are often complex and often challenging to understand because they are theoretical. It is advanced for beginners in this field. Therefore, there is a need for a basic guideline for the learning and testing process in machine learning [1, 2].

In the literature, there is no particular resource in machine learning that examines the training and testing process in detail and offers suggestions. Existing studies usually contain theoretical knowledge [5, 6]. There are no practical explanations in these studies [5–9].

Some of the books in the literature are like a combination of articles [10]. These studies were prepared to classify processes rather than to explain the process in machine learning. In the review studies for machine learning, machine learning applications in a particular area are explained [11, 12]. The functioning of the process and what should be done for this process are not mentioned.

Other studies in the literature are machine learning application studies [13–15]. Some studies are magazines [16], an algorithm developed for machine learning [4, 17]. These studies do not provide guidance on the training and testing process.

This study proposes specific steps for the training and testing process in machine learning. For this purpose, an applied article was prepared as a guide. Section "Machine Learning" provides information about Machine Learning. Chapter "Preparation of DatasetsDataset for Training and Testing Process" describes how to prepare the datasets for the training and testing process. Section "Sampling Methods" describes methods of how to select samples in the training and testing process. Chapter "Feature Selection/Sorting Algorithms" describes the methods to be used in the training and testing process. Section "Feature Selection/Sorting Algorithms" presents the methods to be used in the training and testing process to determine the features. Section "Sample Application Design" presents the application. The application shows how the methods are selected and the results. Section "Results" presents the results of the application. This article experimentally determines which methods can be used in what situations. Section "Discussion and Conclusions" presents the evaluation of the article.

2. Machine Learning

The algorithm is a set of steps to accomplish a task. However, if there is no algorithm available, machine learning algorithms are used to solve the problem. Machine learning algorithms decide from available data regardless of the algorithm. It is the most significant advantage of machine learning algorithms to be independent of algorithm structures. Machine learning is a subset of artificial intelligence. Machine learning has three different working structures. These are supervised, unsupervised, and reinforcement learning [18]. The problem is examined when deciding which machine learning algorithm to use. The method should be selected according to the problem. Otherwise, the operation cannot be performed.

2.1. Supervised Learning. There is a trainer in supervised learning. The trainer is datasets. We can think of the datasets as the numerical form of the experiences of the problem. It contains information about the problem in the matrix. Blood tests are indications of diseases. According to the blood values, the table created with the diagnostic label (Patient/Healthy) is an example of a kind experience matrix.

There are many machine learning algorithms with a supervised learning structure [19, 20]. We can collect them in four headings in general. These are classification algorithms, deep learning, deep transfer learning, and regression methods [18, 21].

2.2. Classification Algorithms. If the dataset has labels like Patient/Healthy, Boy/Girl, Cat/Dog, the data can be classified. In this case, classification algorithms must be selected. There are many classification algorithms. Multilayer Feed Forward Neural Networks (MLFFNN), k-Nearest Neighborhood Algorithm (kNN), Support Vector Machines (SVMs), and Decision Trees (DT) are some of these algorithms [22]. The use of these methods is quite easy on the Matlab platform [23].

2.3. Deep Learning Neural Networks. Deep Learning Neural Networks (DLNN) is an advanced version of MLFFNN [24–26]. In this structure, everything including feature extraction, size reduction, and filtering processes is done automatically [27, 28]. It is often used in image processing [24]. However, if a signal can be converted into an image, DLNN can be used comfortably [14, 29]. DLNN and communication signal applications are also available [19, 22, 30, 31].

2.4. Deep Transfer Learning. In some cases, existing data may not be sufficient to classify with machine learning algorithms. Collecting new data can result in loss of time and money. Besides, if the data is too high and the process of machine learning takes too long, the Transfer Learning method can be preferred [32, 33].

Deep Transfer learning has large data that has been previously prepared [33]. The system works by updating the weight values according to the data. This method is a solution for classification, clustering, and regression problems. There are many applications in the literature [14, 34, 35]. There are several Transfer Learning methods in Matlab, such as AlexNet [36] and GoogleNet [37] in the literature. Specific tasks can be defined in transfer learning by using pretrained models. AlexNet and GoogLeNet are the starting point of the models for specific tasks.

2.5. Deep Transfer Learning with Joint Adaptation Networks. There are also different structures of Deep Transfer learning [38–40]. The most known of these structures is Deep

Transfer learning with joint adaptation networks (JAN) [38]. JAN that learns a transfer network by aligning the joint distributions of multiple domain-specific layers across domains is based on a joint maximum mean discrepancy (JMMD) criterion. This structure uses the adversarial training strategy to maximize JMMD. Stochastic gradient descent performs learning in linear time using the gradients computed.

2.6. Regression. Regression algorithms are a machine learning algorithm used when data labels are numerical [18, 21]. It is frequently used in applications such as exchange rate and stock estimates. Many regression algorithms such as MLFFNN, Support Vector Regression (SVR) [2], Kernel Ridge Regression (KRR) [2], and Gaussian Process Regression (GPR) [18] are available.

2.7. Unsupervised Learning. In this method, the machine learning algorithm uses the data as unlabeled [2, 18, 21]. The algorithm clusters data. The algorithm classifies certain features within the dataset [2, 18, 21]. In this way, the algorithm reveals structures that are hidden in the datasets.

2.8. Clustering Algorithms. Clustering algorithms are the leading methods of unsupervised learning [2, 18, 21]. The user determines how many clusters the data should be separated into. The algorithm distributes the data for the desired N cluster. The clustering algorithm distributes the data to the clusters according to their similarities. There are several clustering algorithms in the literature, such as k-means and k-medoids, hierarchical clustering, hidden Markov models, self-organizing maps, and fuzzy c-means clustering [18, 21].

2.9. Training and Testing Process. The most critical factor affecting the success of machine learning is the training and testing process. An effective training process improves the quality of the developed system (Figure 1). Researchers divide datasets into two parts for training and testing. However, the separation process is done according to specific rules. These are described in detail in section "Sampling Methods." The amount of training and test is the most critical factor in the success rate. If there is a high correlation between the features and the label, the Training-Test set is divided by 50%–50%. This means that 50% of all the data will be used for training and 50% for the test. However, if there is a fear of success falling, the rate of training can be increased. The training-testing ratio used in the literature varies according to the data structure (Table 1). Less than 50% of the training data is not preferred because the test results will be negatively affected.

After the machine learning model is trained according to the training data, it is also tested using the training data. The purpose of this is to determine how much data is learned (Figure 1). Performance evaluation procedures are performed according to specific criteria. These criteria vary according to the structure of the data. Section "Performance Evaluation Criteria" presents the performance evaluation criteria in detail.

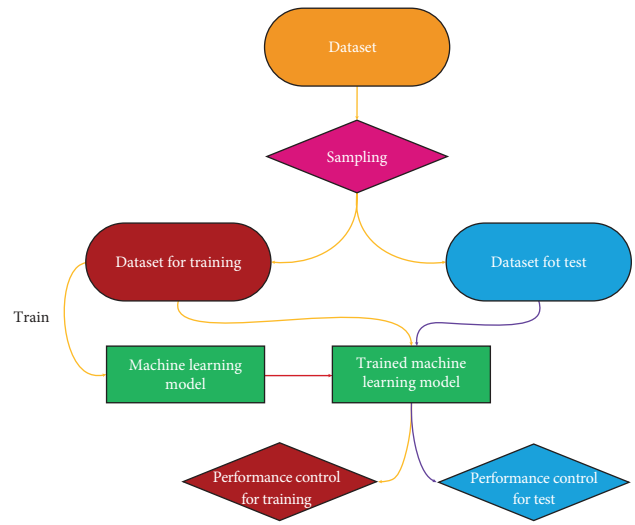

Figure 1: Training and test process in machine learning.

Table 1: The ratio of training and test sets according to the total dataset (%).

Training (%)	Test (%)
50	50
55	45
60	40
65	35
70	30
75	25
80	20
85	15
90	10
95	5

Once the training process is completed, the machine learning model tested with test data has never been seen before. The researcher evaluates the test performance according to the performance evaluation criteria (section "Performance Evaluation Criteria"). The research can be repeated by changing the training and test data in the training and testing process to avoid the situation of unstable data [41]. In this case, the researcher uses the average of performance values.

2.10. Performance Evaluation Criteria. Performance evaluation criteria vary according to the data structure and method. If data labels have categorical variables (such as Patient/Healthy), classification performance criteria (section "Performance Criteria for Classification") are used. Regression performance criteria (section "Performance Criteria for Regression") are used if the data labels are continuous numerical variables (such as 0.1, 2, 3, and 1.1).

3. Performance Criteria for Classification

These performance criteria are used when data labels are categorical variables. Class labels must have at least two

labels. This work explains the performance evaluation of two labeled data. There are many performance evaluation criteria in the literature [42]. Classification studies can be examined to find more performance evaluation criteria [13, 15].

Since the performance evaluation criteria used for classification problems are trendy, there are equations in many articles. Therefore, detailed equations are not given in this article.

The following performance evaluation criteria are used most frequently for classification problems in the literature. (1) Confusion Matrix, (2) Accuracy rate (%), (3) Sensitivity or True Positive Rate, (4) Specificity or True Negative Rate, (5) Kappa Value, (6) ROC Curve, (7) AUC, (8) k-fold Crossvalidation, (9) Leave-one-out Crossvalidation, and (10) Correlation Coefficients.

3.1. Performance Criteria for Regression. Regression tries to explain the relationship between variables with mathematical equations [43]. In machine learning, this process is done secretly by machine learning algorithms, not by the help of equations. A variable enters the machine, and the machine reveals results. Some parameters are calculated to interpret the regression results.

3.1.1. Correlation Coefficients. The correlation coefficient method is selected according to the variable types in classification (Table 2). Because the predicted variables are continuous variables, Pearson (r) or Sperman (r_s) correlation coefficient formulas are used. When selecting these two calculation types, it is checked whether the data is distributed normally. If the data has a normal distribution, the Pearson correlation coefficient formula is used. If it does not show normal distribution, the Spearman Correlation Coefficient (r_s) is used, which is the nonparametric equivalent of the Pearson correlation coefficient.

3.1.2. Relationship between Correlation and Estimation. If the correlation between the two variables is $|r| < 0.70$, the estimation error rate of the system will be quite high. If $0.5 < |r| < 0.70$, the estimation of the system is low. If the value is $0.7 < |r| < 0.90$, the estimation of the system is moderate. If $0.9 < |r|$, then the estimation of the system is high [43].

3.1.3. Raw Wastes (e_i). Raw wastes are the difference between actual values (t_i) and estimated values (y_i) (equation (1)). As e_i approaches zero, the developed machine learning-based system is so successful. As A approaches zero, the developed machine learning-based system is so successful [43]:

$$e_i = t_i - y_i. \tag{1}$$

3.1.4. Standard Error (s). s indicates the compliance of the developed method with the data (equation (2)) [43]. When the correlation is less than 1, the system may not predict with 100% accuracy. In this case, deviations (e_i) from the actual values occur. The standard error (s) of the proposed system is the standard deviation of deviations. As e_i decreases, s decreases. System reliability increases as s decreases [43]:

$$s = \sqrt{\frac{\sum_{i=1}^{n} (t_i - y_i)^2}{n-2}} = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n-2}}. \tag{2}$$

Table 2: Feature selection algorithms.

Feature selection algorithms			
Sorting algorithms		Selection algorithms	
Name	Ref	Name	Ref
Chisquare	[44]	Brogreg	[45]
Fisher	[46]	FCBF	[47]
Information gain	[48]	MRMR	[49]
Kruskal wallis	[50]	MRMR information	[49]
Gini out	[51]	MRMR parson	[49]
Relief F	[52]	SBMLR	[53]
T test			

FCBF: Fast Correlation-Based Filter Solution, MRMR: Max-Relevance Min-Redundancy, SBMLR: Sparse Multinomial Logistic Regression via Bayesian L1 Regularization.

3.1.5. Explanatory Coefficient (R^2). The explanatory coefficient defines the percentage of the change in the total change that can be explained by the regression model (equation (3)) [43]. In other words, R^2 indicates that the independent variable can explain the percentage of total changes in the dependent variable.

$$R^2 = r^2 = \frac{\mathrm{KT}_R}{\mathrm{KT}_Y}. \tag{3}$$

3.1.6. R^2 Explanatory Equations. Real values (t_i), Estimated values (y_i), Number of data (n), Sum of Squares T KT_T, Sum of Squares R (KT_R), Sum of Squares A (KT_A):

$$\begin{aligned} \mathrm{KT}_T &= \mathrm{KT}_R + \mathrm{KT}_A, \\ \mathrm{KT}_T &= \sum_{i=1}^{n} (t_i - \overline{t})^2 = \sum_{i=1}^{n} t_i^2 - \frac{(\sum_{i=1}^{n} t_i)^2}{n}, \\ \mathrm{KT}_R &= \sum_{i=1}^{n} (y_i - \overline{t})^2, \\ \mathrm{KT}_A &= \sum_{i=1}^{n} (t_i - y_i)^2. \end{aligned} \tag{4}$$

3.1.7. MSE. MSE refers to the mean of the squares of errors (equation (5)). The e_i in the equation expresses the errors and is calculated according to the following equation:

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} e_i^2. \tag{5}$$

3.1.8. RMSE. RMSE refers to the square root of MSE. As MSE and RMSE approach zero, the error rate decreases (equation (6)).

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2}. \quad (6)$$

3.1.9. Error Rate. Error Rate is the percentage indicator of the change between the actual value and the estimated value:

$$H_{\text{err}} = \frac{1}{n}\sum_{i=1}^{n}\frac{|t_i - y_i|}{t_i} \times 100. \quad (7)$$

4. Proposed Preparation of Datasets for Training and Testing Process

One of the essential materials in machine learning is the data matrices. The manner in which these matrices are prepared influences system performance. This process must be carried out meticulously. Otherwise, the study may need to be repeated because the results are not good enough.

The machine learning process is done with matrices. Therefore, the data must be prepared as matrices. Table 3 shows the sample matrix for disease diagnosis. The values in the "Blood Test Parameters" column in the matrix are features for machine learning. In other words, matrices are input data for machine learning. Individual number is for informational purposes only. This column is not given to machine learning. The doctor fills the label column according to the health status (Health/Patient) of the individual. For example, individual 7 has Z Blood Test Parameters. The doctor gave the patient a "1-Healthy" label. This process is repeated for other individuals. Here, it is important to note that the data should be distributed in a balanced way. Balanced data means that the number of patients and healthy individuals are close to each other. In this case, there are 500 patients, 500 healthy individuals, and 1000 individuals in total (Table 3). Section "Special Case: Unbalanced Datasets" provides detailed information for unbalanced data.

Machine learning aims to be able to identify the doctor's diagnosis more quickly. However, when performing this procedure, the characteristics that the doctor cannot consider (Blood Test Parameters) should be revealed. For example, while the doctor diagnoses according to many parameters, machine learning can determine a different parameter for this disease.

Once the data is prepared in this way, the data is ready for the training and testing process. However, these matrices should be prepared carefully.

4.1. Special Case: Unbalanced Datasets. Unbalanced data is part of our daily lives. The collected data may not always be balanced. In this case, two different methods are applied. Assume that group 1 has m sample and group 2 has n sample ($m > n$). In the first method, n samples are selected from group 1. In this case, the data is balanced. Table 4 shows a balanced and unbalanced matrix. This sample reduction process is called subsampling. Section "Sampling Methods" presents the subsampling in detail.

Data can be balanced by sampling methods when the number of samples is sufficient. However, if the number of data is insufficient, methods such as Boosting and AdaBoost can be used [17, 54]. In these methods, each sample takes part in training and testing processes. The datasetdataset is divided into n parts. The $n-1$ piece takes part in the training stage of n classification. Each piece takes part in the testing phase. The performance is calculated by taking the average of the accuracy rates obtained from each classification process.

5. Sampling Methods

Sampling is the process of creating subsets (n) by selecting samples from the universe n. The new cluster has properties of the universe ($N > n$). Samples can be selected from the universe with different methods. Methods developed to select samples from the universe are called sampling methods.

The sampling process is essential for machine learning training and testing stages. An unbalanced sampling of data can directly affect training and test performances. Therefore, both training and test data should represent the entire datasetdataset. Figure 2 shows graphically correct and incorrect sampling. Sampling, as in Figure 2 (Red), is not made only from a certain area. The new cluster (Green) also includes two groups in a balanced manner.

If the X matrix is [1 2 3 4 5 6 7 9 10], we can do the sampling for training and testing as follows. [1 3 5 7 9] sets for training and [2 4 6 8 10] sets for testing can be selected. The training set should cover the total datasetdataset. Otherwise, the results will be unstable. If [1 2 3 4 5] is selected for training, when the number 10 data is tested, probably machine learning gives the wrong answer. Because the value 10 is an extreme value according to the training set.

This section discusses sampling methods that can be used in the training and testing phase. These methods are mainly the subject of statistics. However, due to the similarity of the problems (Problem: Sampling), sampling methods are needed in the field of machine learning.

5.1. Simple Random Sampling. A simple random sampling method randomly selects samples from the datasetdataset [55]. In order to select a sample, the samples are numbered sequentially up to N. In the second step, n random integers are determined. n is the number of samples to be selected. The Kendall or Smith, Random Numbers table, is used to determine random numbers. The same integer must not be selected twice. The selected number x should not be greater than N.

We can apply a simple random sampling method to machine learning in the following way. First of all, how much data will be used for training and how many samples are selected. The remaining samples are used during the testing phase. However, this method is not suitable for the

Table 3: Sample machine learning matrix for disease diagnosis.

Subject No	Blood test parameters									Label
	A	B	C	D	E	F	G	...	Z	
1	123	0.79	6.40	6.98	62.14	0.51	203.93	...	12.89	1
2	184	3.16	1.81	4.07	39.69	1.05	365.96	...	14.52	2
3	86	2.26	0.45	8.71	50.11	0.84	236.41	...	14.65	1
4	12	1.30	7.23	9.93	131.63	0.68	224.03	...	12.12	2
5	168	3.42	3.47	3.94	183.35	0.94	165.32	...	12.35	1
6	122	2.34	6.61	9.61	89.71	1.15	137.70	...	14.37	2
7	69	0.71	3.84	3.62	98.70	0.92	192.67	...	13.39	1
8	193	3.23	6.27	7.94	190.50	0.87	317.83	...	14.12	2
9	44	4.57	0.22	8.59	184.29	0.92	334.86	...	12.18	1
10	126	2.35	9.11	6.97	141.26	0.82	308.14	...	11.89	1
11	117	4.25	8.01	9.25	191.13	0.22	102.94	...	15.15	2
12	75	2.76	7.46	9.34	155.69	1.11	352.96	...	14.83	2
13	28	1.26	8.13	7.57	86.87	1.49	376.70	...	15.67	1
14	5	4.12	3.83	4.40	138.98	0.60	331.29	...	11.54	1
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...
1000	20	1.32	4.28	4.92	72.07	1.459762	112.798	...	11.91	2

m = 500 Number of patients, n = 500 Number of healthy, T Total Number of Patients, $T = m + n = 500 + 500 = 1000$ 1 Healthy, 2 Patient.

Table 4: Unbalanced and balanced data matrices.

Matrix	Matrix 1		Matrix 2			Matrix 3	
	Class 1	Class 2	Class 1	Class 2	Class 3	Class 1	Class 2
Unbalanced	175	50	200	75	60	1500	1200
Balanced	50	50	60	60	60	1200	1200

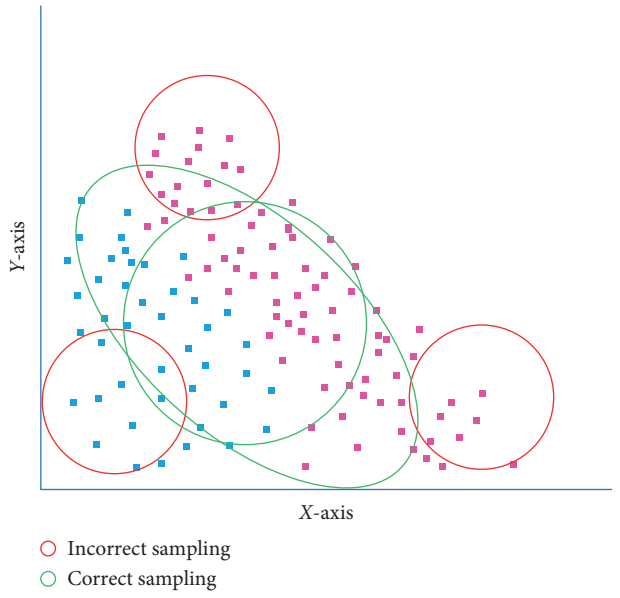

Figure 2: Ideal sampling.

training and testing phase. Because the data must be balanced on the set of training and testing; if 100 patient records are used in training, 100 healthy records should be used. The balance should be provided between groups.

5.2. Systematic Sampling. As with simple random sampling, samples are numbered from 1 to N. The features to be selected are determined according to equation (8). In the equation, d is the number of cycles, and a is the starting sample number. The user determinates d according to $d \leq N/n$.

$$a, a + d, a + 2 \times d, + \ldots + a(n-1) \times d. \quad (8)$$

5.3. Stratified Random Sampling. Stratification is the process of grouping before sampling [55]. In machine learning, groups are labels such as Patient/Healthy. The n sample to be selected must represent the N-instance cluster. For this, we need to do stratification sampling.

In this method, the samples are divided into groups based on tags. Then, n the sample is selected according to the simple random sampling method. In this way, the label distribution within the training data will be balanced.

5.4. Stratified Systematic Sampling. Stratification is the process of grouping before sampling [55]. Then, n the sample is selected according to the Systematic Sampling method. In this way, the label distribution within the training data will be balanced.

In machine learning, the preferred method of Stratified Random and Systematic Sampling will be more suitable for a balanced distribution of data.

5.5. Special Case: Dataset Shift in Machine Learning. Dataset shift is a common problem that occurs in machine learning training and testing processes. [56]. The basis of the problem is that the data chosen for the training and testing process do not show a similar distribution. Figure 3 shows the difference between the training and test dataset distribution.

In such cases, the trained network gives abnormal answers to the tests. This situation reduces the test success rate.

5.6. Special Case: What You Saw Is Not What You Get: Domain Adaptation Using Asymmetric Kernel Transforms. This problem occurs when the data in the training set are collected to specific standards [57]. When real-world data is used for testing, the accuracy of the test data is significantly reduced, as the test data does not comply with the training set standards. Conversion of the dataset can be done for the solution to this problem. The most known transformation process is the ARC-t method [57]. The method provides a flexible model for supervised learning of nonlinear transformations between domains [57].

6. Feature Selection/Sorting Algorithms

In Table 3, "Blood Analysis Parameters" is a feature for machine learning. The number of properties has positive and negative effects on machine learning performance. The general purpose of feature selection algorithms is to select features that will improve performance. There are many feature selection criteria in the literature.

6.1. Feature Selection Algorithms. There is two different feature selection algorithm structure in the literature (Table 2). The first is the feature selection algorithms that selects the features according to specific criteria. The selection criteria of the methods may vary. The selected feature amount may be different for each algorithm.

The number of features selected may be criterion-based or may be set as a standard by the programmer.

6.2. Feature Sorting Algorithms. Feature sorting algorithms makes sorting without selecting properties (Table 2). This order is from the most relevant to the least relevant, according to the level of relationship with the tags of the properties. After sorting, the user can select the desired amount of features. The reason why features are not initially selected is that the number of properties in each data will have a different rate of performance.

7. Sample Application Design

The purpose of the application is to determine which methods should be used in the machine learning training and testing process. This example explains how to choose which method is to be selected from the first step to the last step. In this exemplary embodiment, from the first step to the last step, which methods will be selected at which stage will be explained. The study will be carried out step by step according to the flow diagram in Figure 4.

7.1. Materials and Methods. This section will examine how different methods affect machine learning training and testing.

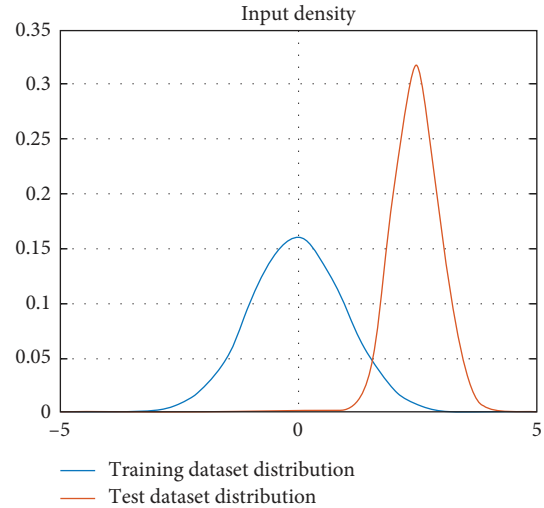

Figure 3: Training and test datasets distribution.

7.2. Sample Datasets. Four different datasetdatasets were used in the study [58–60]. Their distribution is given in Table 5. The experiment sets were especially balanced. Because unbalanced data classification requires a very complex process. Therefore, these sample sets will be used for a healthier example.

7.3. Machine Learning Algorithms. Three classifiers were used for the classification process. These are the Decision Tree (DT), k-Nearest Neighbors Algorithm, and Support Vector Machines (SVMs).

7.4. Feature Sorting Algorithms. In practice, Fisher Feature Sorting Algorithm described in section "Feature Sorting Algorithms" was used.

7.5. Performance Evaluation Criteria. The application is a classification application. The parameters in section "Performance Criteria for Classification" are used for this. Accuracy rate (%), Sensitivity, Specificity, and number of traits are calculated for accuracy rate.

8. Results

The study aims to determine the processes affecting the education and testing process in machine learning and to choose the appropriate methods for this process. In the article, a detailed explanation is made for the machine learning training and testing process. Then a sample application was designed, and the results were obtained. In this application, the effect of sampling methods on performance, the effect of training and test rates on performance, the effect of classifiers on performance, the effect of datasetdatasets on performance, and the effect of feature selection algorithms on performance were investigated. As a result of this review, it has been tried to determine which methods can be selected at each step.

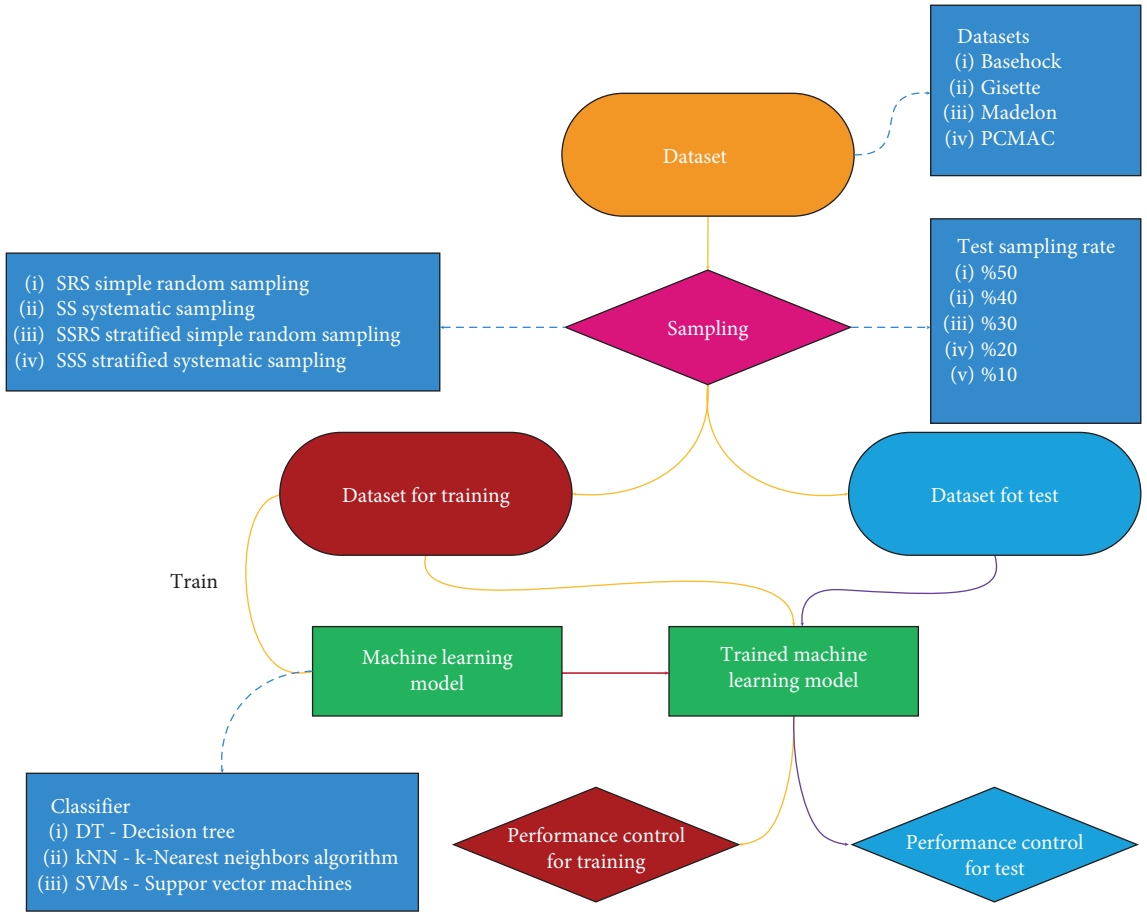

Figure 4: Training and testing process in machine learning.

Table 5: Datasets and distributions used for performance testing.

Datasets	Features	Number of class		Total	Ref
		Class 1	Class 2		
Basehock	4862	994	999	1993	[58, 59]
Gisette	5000	3500	3500	7000	[59]
Madelon	500	1300	1300	2600	[59]
PCMAC	3289	982	961	1943	[58, 59]

Four different sampling methods were used in the study. These are simple random sampling, systematic sampling, stratified simple random sampling, and stratified systematic sampling (Table 6-Method). Data were classified after sampling for training and testing (Figures 5–8).

With a simple and systematic sampling method, datasetdatasets for training and testing were created unevenly (Table 6). In these methods, unbalanced data were selected from Class 1 and Class 2 because the selection was made without segregation. Training-C1 and Test-C1 must be equal when Training and Tests are 50%. However, in simple and systematic sampling, these numbers are unstable for each datasetdataset. In the Simple and Systematic Sampling with Layer, Education-C1 and Test-C1 are equal. This is the desired situation in machine learning. The values of C1 and C2 in each training-test set group are the number of samples selected by stratified sampling methods. Deviations in these numbers can change the performance of training and testing.

When the training-test sets are selected with simple and random sampling methods, performance values change by chance (Figure 5, kNN-Simple Blue, DT-Systematic Pink). The sudden changes in performance are the most significant indicator of the method's instability (Figure 6, DT-Simple Blue--Figure 7, DT, kNN-Simple Blue, Systematic Pink--Figure 8, DT-Simple Blue).

In the stratified sampling methods, the number of classes (C1 and C2) is fixed in the training and test datasetdatasets. Changing is only the selected example. The balance of numbers positively affects performance. Increasing the test percentage decreases test performance slightly. When the graphs are analyzed, the performance chart from 10% to 50% is balanced for simple stratified and systematic sampling methods (Figures 5–8).

In the stratified simple sampling, each time a different sample selection is made. Therefore, a performance change may occur (Figure 6, kNN-Stratified Simple Black). The samples to be selected in the stratified systematic sampling are given by the formula (section "Stratified Systematic Sampling"). Therefore, stratified systematic sampling has a better and balanced performance compared to the Stratified

Table 6: Datasets and distributions used for performance testing.

Basehock dataset																				
Percent	50%		50%		60%		40%		70%		30%		80%		20%		90%		10%	
Method	Train		Test		Train		Test		Train		Test		Train		Test		Train		Test	
	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2
SRS	505	492	489	507	607	589	387	410	697	699	297	300	779	816	215	183	892	902	102	97
SS	478	517	516	482	592	602	402	397	689	705	305	294	790	803	204	196	888	904	106	95
SSRS	497	500	497	499	597	600	397	399	696	700	298	299	796	800	198	199	895	900	99	99
SSS	495	498	499	501	595	598	399	401	694	698	300	301	794	798	200	201	893	898	101	101
Gisette dataset																				
Percent	50%		50%		60%		40%		70%		30%		80%		20%		90%		10%	
Method	Train		Test		Train		Test		Train		Test		Train		Test		Train		Test	
	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2
SRS	1704	1796	1796	1704	2099	2101	1401	1399	2433	2467	1067	1033	2789	2811	711	689	3144	3156	356	344
SS	1731	1767	1769	1733	2097	2101	1403	1399	2437	2461	1063	1039	2788	2810	712	690	3152	3146	348	354
SSRS	1750	1750	1750	1750	2100	2100	1400	1400	2450	2450	1050	1050	2800	2800	700	700	3150	3150	350	350
SSS	1748	1748	1752	1752	2098	2098	1402	1402	2448	2448	1052	1052	2798	2798	702	702	3148	3148	352	352
Madelon dataset																				
Percent	50%		50%		60%		40%		70%		30%		80%		20%		90%		10%	
Method	Train		Test		Train		Test		Train		Test		Train		Test		Train		Test	
	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2
SRS	636	664	664	636	767	793	533	507	927	893	373	407	1041	1039	259	261	1167	1173	133	127
SS	649	649	651	651	786	772	514	528	910	908	390	392	1038	1040	262	260	1169	1169	131	131
SSRS	650	650	650	650	780	780	520	520	910	910	390	390	1040	1040	260	260	1170	1170	130	130
SSS	648	648	652	652	778	778	522	522	908	908	392	392	1038	1038	262	262	1168	1168	132	132
PCMAC dataset																				
Percent	50%		50%		60%		40%		70%		30%		80%		20%		90%		10%	
Method	Train		Test		Train		Test		Train		Test		Train		Test		Train		Test	
	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2	C1	C2
SRS	506	466	476	495	593	573	389	388	696	665	286	296	796	759	186	202	887	862	95	99
SS	489	481	493	480	585	579	397	382	688	671	294	290	781	772	201	189	872	875	110	86
SSRS	491	481	491	480	590	577	392	384	688	673	294	288	786	769	196	192	884	865	98	96
SSS	489	479	493	482	588	575	394	386	686	671	296	290	784	767	198	194	882	863	100	98

SRS: Simple Random Sampling, SS: Systematic Sampling, SSRS: Stratified Simple Random Sampling, and SSS: Stratified Systematic Sampling.

Simple and Stratified Systematic sampling method (Figures 5–8).

As the total percentage was set to %100, the training set was set between 50 and 90% and the test set was set between 50 and 10% and classified (Table 6, Figures 5–8).

Since the Stratified Systematic Sampling method is more balanced than other sampling methods, we will refer to this method comparing the performance of Training and Test rates. When the test rate increases, the accuracy is expected to decrease (Figure 6, kNN-Stratified Systematic Red). However, this does not always happen because there is a difference between the classifiers (Figures 5–8, Stratified Systematic Red). In fact, while the test rate increases in a dataset, performance decreases while performance increases in the other dataset (Figure 6 SVM, Stratified Systematic Red--Figure 7 SVM, Stratified Systematic Red).

In order to determine the effect of the classifiers on the performance, three different classifiers and the different datasets (4), different rates (5), and different sampling methods (4) and selected training and test data were classified (Figures 5–8).

Different classifiers can adapt to a dataset differently under the same conditions. It can have different performance values (Figure 5, DT, kNN, SVMs, Stratified Systematic Red). A classifier may show a low performance (Figure 6 DT, Stratified Systematic Red) for a dataset, while the best performing classifier for another dataset (Figures 7 and 8 DT, Stratified Systematic Red). The classifiers under the same conditions showed different performances in the same datasets (Figures 5–8, Stratified Systematic Red).

Four different sets of data have been classified (Figures 5–8) in order to examine the effect of datasets on performance in training and testing processes. In each classification process, different classifiers and different sampling methods and different Training-Test rates were used. In this way, a detailed analysis was made.

When the same datasets are classified under the same conditions as different classifiers, it is impossible to achieve the same result. As with personal differences, the structure of classifiers is different. Therefore, each classifier cannot fit perfectly into each dataset.

The best classification performance for the Basebock dataset belongs to DT, while the worst classification performance belongs to kNN (Figure 5, Stratified Systematic Red). On the other hand, the best performance for the Gisette dataset was SVMs, while the worst performance was

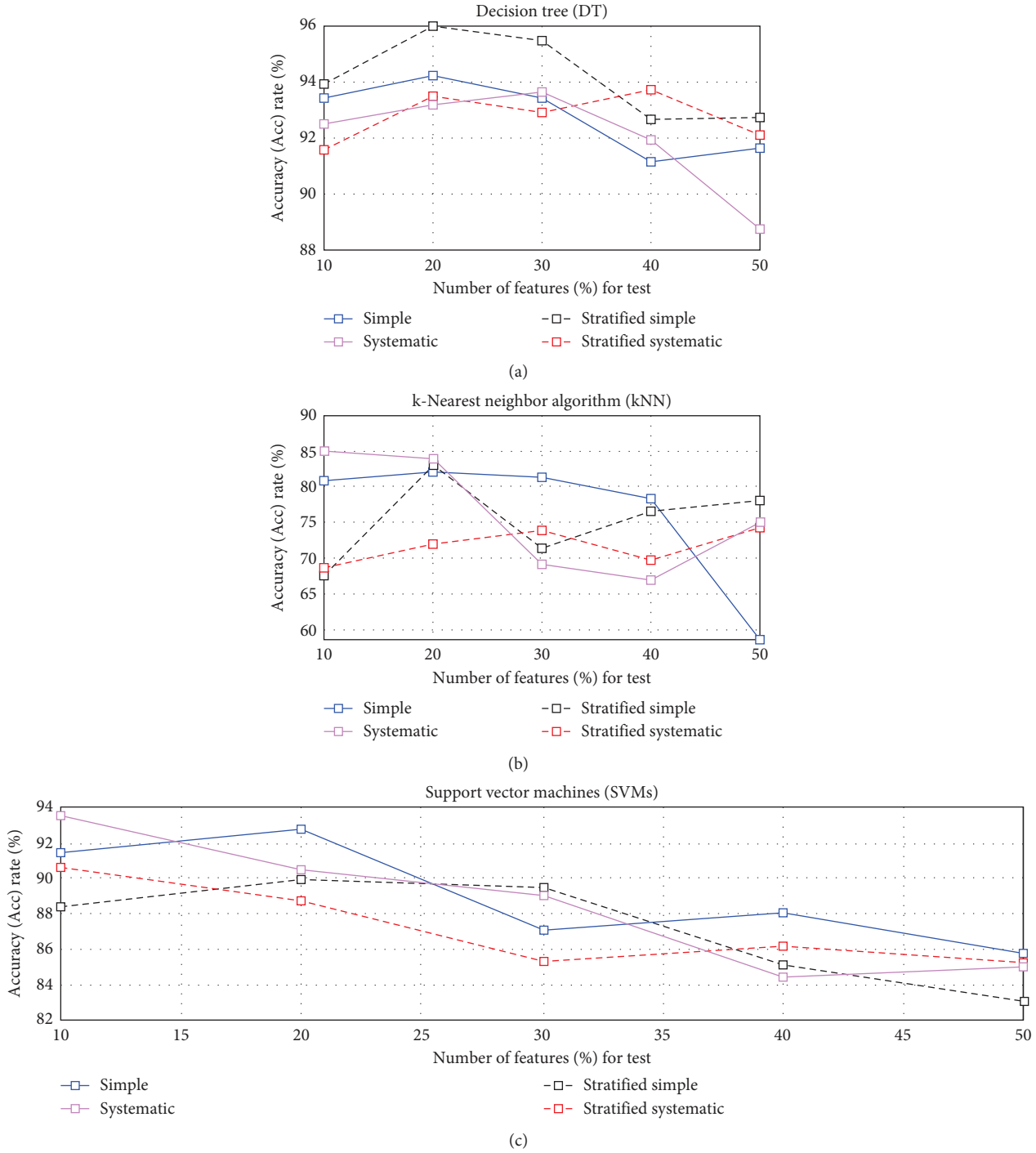

Figure 5: Performance for basehock.

shown in DT (Figure 6, Stratified Systematic Red). This indicates that the classifiers perform differently for different datasets. Even in different training and test percentages, each classifier has different performance. This may be due to classifying the working methods and distribution of data.

Features with the Fisher Feature Sort algorithm are sorted by interest level with tags. For each dataset, 5–50% of all features are selected and classified by classifiers (Figure 9). The training and test dataset is divided according to the stratified systematic sampling theorem.

Classifier performances vary according to the selected feature quantity (Figure 9). For the four different sets of data, the optimum range is 20–25%.

The classifier performances perform differently for the same dataset (25%) (Figure 9). While a classifier is the best in a dataset (Figure 9, Basehock DT), it is worst in the other dataset (Figure 9, Gisette DT).

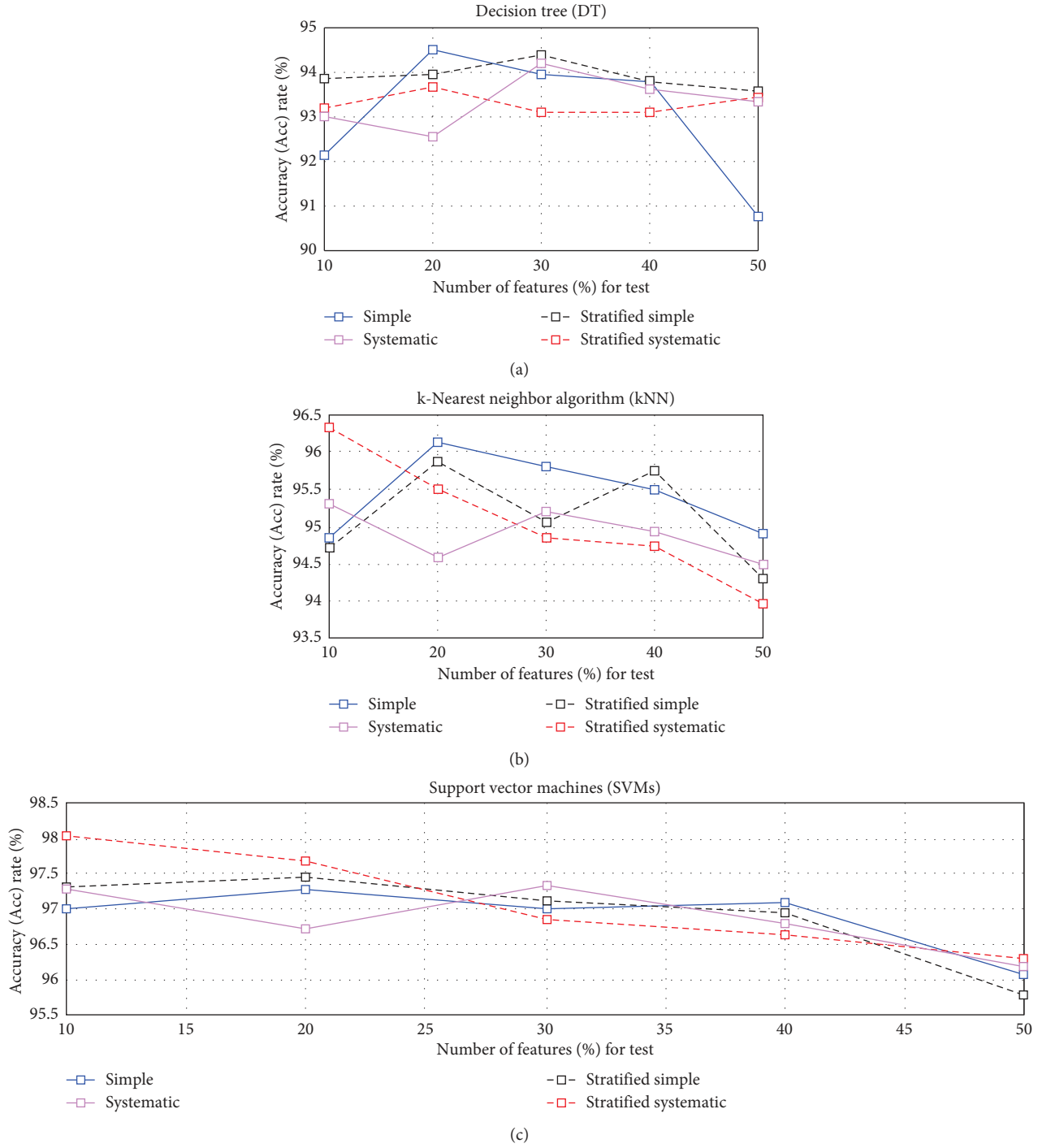

Figure 6: Performance for Gisette.

9. Discussion and Conclusions

The experimental results obtained in the study will be evaluated under 4 different headings. The aim of the study is to examine the effect of training and testing process on performance in machine learning. For this, experimental studies were carried out in the steps taken in the training-test process, and the results were evaluated.

The sampling theorem is the primary subject of statistics. Sampling methods are not used in machine learning often. However, it is an issue to focus on, and it affects performance extremely [55]. The major innovation in this paper is to demonstrate the effect of sampling theorems on the learning and testing process in machine learning and to recommend the use in the training and testing process.

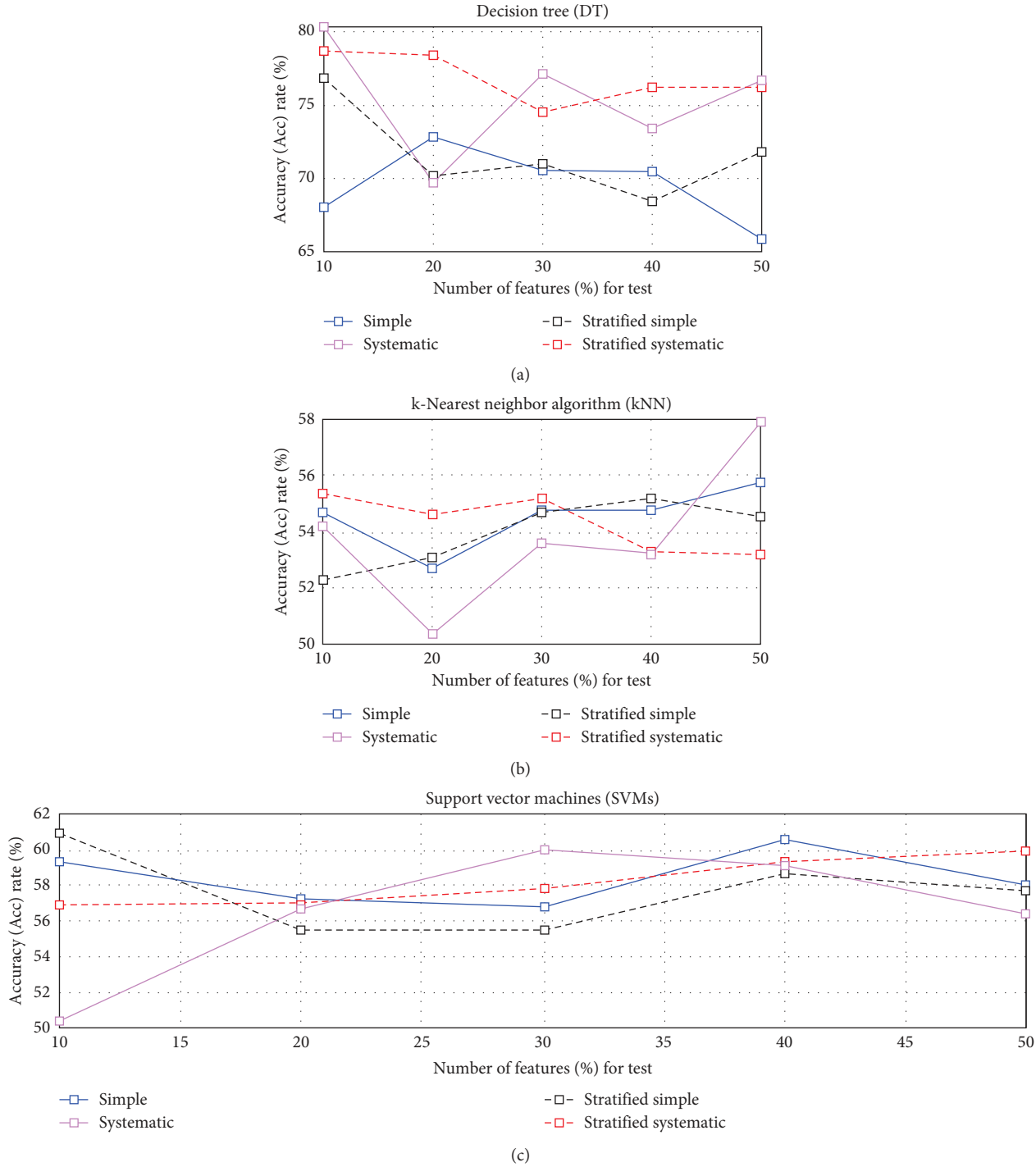

Figure 7: Performance for madelon.

9.1. Effect of Sampling Methods on Performance. In the literature, examples for training and testing are usually randomly selected [5–9]. You are asked to have unused data during the training phase. However, there is no detail on how to choose. However, the sampling theorem is the main subject of statistics and how the samples should be chosen. That is what is said to be about the test [55]. We think that these methods should be used in machine learning.

Sampling methods are used to select data to be used in the training and testing process. The selected samples are asked to represent the entire dataset while selecting the sample for the training from the entire dataset. Therefore, a stable sampling method should be used. It cannot be said that a method that produces different results in every process works stable. Simple random sampling can be considered as an unstable sampling method because it selects different samples each time.

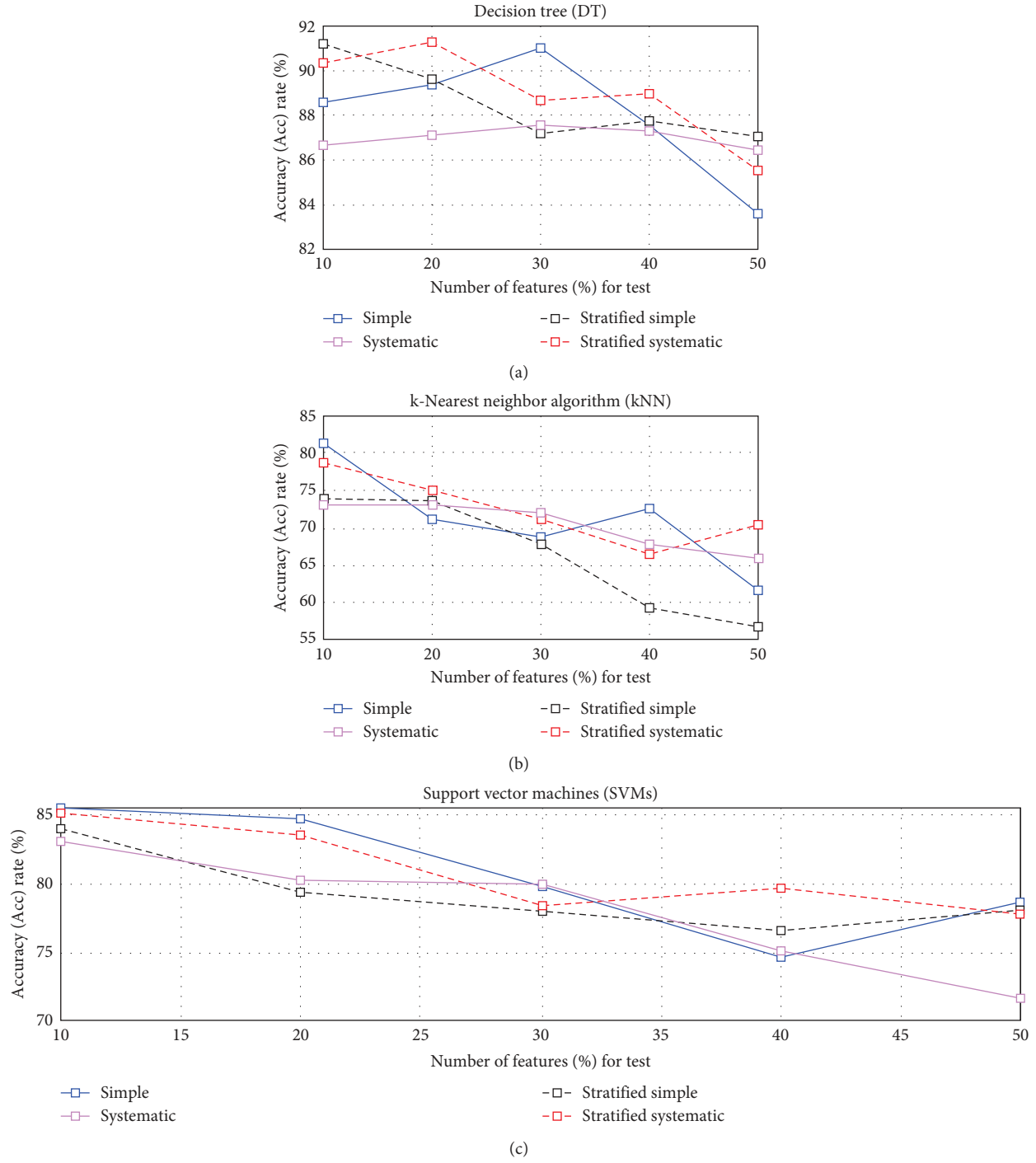

Figure 8: Performance for PCMAC.

In the training and testing process, the datasets should be divided into classes first. Stratified sampling is therefore required. Unbalanced data selection between classes may affect the performance of the training and testing process positively or negatively. However, we want a stable system. Therefore, a stratified-based sampling method is needed. Although stratified, random sample selection cannot stabilize the system. In this case, the selection of samples according to certain rules can be used to create a stable training and testing process in each dataset. In the study, different classifications were tested with different classifiers in different datasets, where a stratified systematic sampling method was employed. Therefore, it is recommended to use a method that can be strictly determined in the training and testing process.

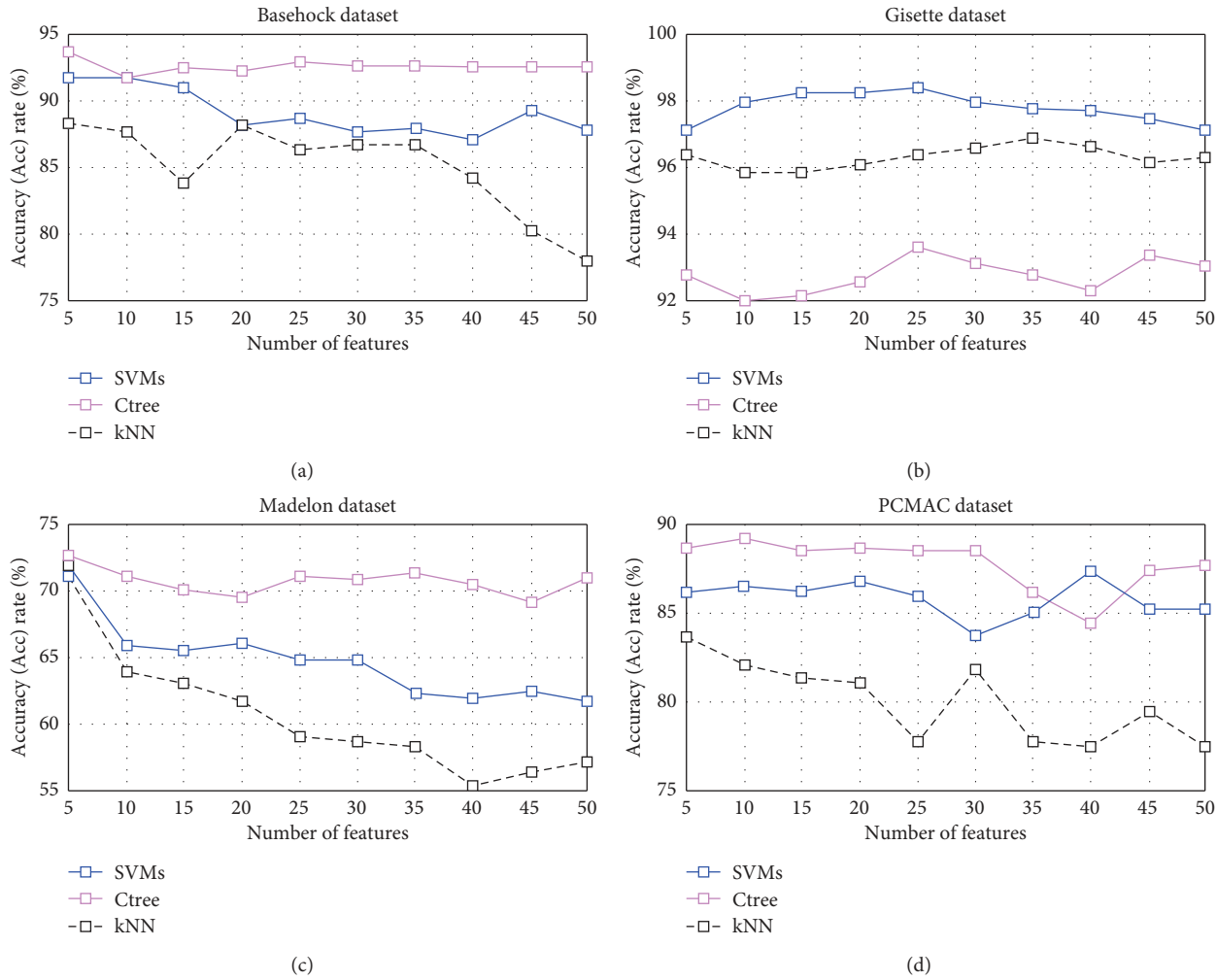

Figure 9: Performance for Fisher.

9.2. Effect of Training and Test Rate on Performance. If the developed system is desired to be tested under the most difficult conditions and the number of data is high, the test rate can be selected as 50% [13]. However, if the trust in dataset is low or the number of samples in the dataset is low, the test rate can be selected between 10 and 20%. In general, the test rate is between 20 and 50%. We see that these ratios are appropriate according to the experimental results.

Excessive data selection for education can result in memorization [1]. In order to prevent this, it is recommended that more than 90% of the samples should not be selected for training.

9.3. Effect of Classifiers on Performance. Classifier selection is very important [13]. A classifier in the same dataset shows an accuracy of %50, while another classifier can show an accuracy of 80%. The use of more than one classifier to solve a problem in order to prevent these situations is important in order to determine the actual performance of the study.

9.4. Effect of Datasets on Performance. The harmony within the dataset is important. The high correlation of each feature with the class labels is an indication of compliance. However, this fit may not always be achieved. Although features in the dataset have low correlation, many features can improve performance in machine learning when combined [13].

When you collect data for your research, the compatibility of these data may be low. Therefore, the classification performance may be low. However, in order to improve performance, in this case, the selection of other methods described in this article should be made carefully. If the compliance within the dataset is low, the test percentage ratio can be drawn up to 10%.

If the number of features is sufficient, the performance can be increased with feature selection or feature sorting algorithms. In addition, different classifiers can be used because each classifier may not match your dataset. You may need to try to find the appropriate one.

9.5. Effect of Feature Sorting Algorithm on Performance. Classes may take quite a long time to run if the number of features is high. In this case, the number of features can be

reduced without damaging performance [15, 61]. It may be useful to use the feature selection or sorting algorithm [46]. In this study, the Fisher Feature Sort algorithm is used [46]. With the help of this algorithm, we can determine the number of features we want to select. We can monitor the performance according to the change of the selected property quantity and determine the appropriate number of features according to our dataset. Therefore, this type of feature selection or sorting algorithms should be used to improve performance. Performance should be evaluated not only in terms of accuracy but also in terms of reducing workload.

In some cases, even if the feature is selected, the accuracy rate may not increase. However, it should be considered that the number of features is reduced in this case. In fact, the performance has increased. How do we express this? When we have 90% accuracy with 1000 features, we have 90% accuracy with 20 properties after feature selection. Note that there is a tremendous reduction in the number of features. In this case, the labor force required for the classifier to work will be reduced. Likewise, the time spent to remove the feature will be reduced. The energy that a system consumes when extracting 1000 properties cannot be compared to the energy it consumes when it extracts 20 properties.

Data Availability

Data are available in https://archive.ics.uci.edu/ml/datasets.php.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors. This article does not contain any studies with animals performed by any of the authors

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This project was funded by the Deanship of Science Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia, under Grant no. D-535-135–1441. The authors, therefore, acknowledge with thanks to DSR technical and financial support.

References

[1] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining-Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, Burlington, MA, USA, 2013.

[2] C. Aldrich and L. Auret, *Unsupervised Process Monitoring and Fault Diagnosis with Machine Learning Methods*, Springer London, Berlin, Germany, 2013.

[3] M. Somvanshi and P. Chavan, "A review of machine learning techniques using decision tree and support vector machine," *in*, in *Proceedings of the 2016 International Conference On Computing Communication Control And Automation (ICCUBEA)*, pp. 1–7, Pune, India, August 2016.

[4] Z. Gong, P. Zhong, and W. Hu, "Diversity in Machine Learning," *IEEE Access*, vol. 7, pp. 64323–64350, 2019.

[5] S. Theodoridis, *Perspective, Machine Learning : A Bayesian and Optimization*, Academic Press is an imprint of Elsevier, London, UK, 2015.

[6] M. Gori, *Machine Learning A Constraint-Based Approach*, Amsterdam, Netherlands, 2018.

[7] B. K. Natarajan, *Machine Learning, a Theoretical Approach*, Elsevier Science, CA, USA, 1991.

[8] H. Aaron and D. Fleet, *Machine Learning and Data Mining: Introduction to Machine Learning Handout*, pp. 1–4, Horwood Publishing, Chichester, UK, 2009.

[9] E. Hunt, "Machine learning: an artificial intelligence approach (vol. 2)," *Journal of Mathematical Psychology*, vol. 31, no. 3, pp. 299–305, 1987.

[10] N. Dey, S. Borra, A. Ashour, and F. Shi, *Machine Learning in Bio-Signal Analysis and Diagnostic Imaging*, Academic Press, Amsterdam, Netherlands, 2019.

[11] J. T. McCoy and L. Auret, "Machine learning applications in minerals processing: A review," *Minerals Engineering*, vol. 132, pp. 95–109, 2019.

[12] B. M. Henrique, V. A. Sobreiro, and H. Kimura, "Literature Review: Machine Learning Techniques Applied to Financial Market Prediction," *Expert Systems with Applications*, vol. 124, 2019.

[13] M. K. Uçar, M. R. Bozkurt, C. Bilgin, and K. Polat, "Automatic detection of respiratory arrests in OSA patients using PPG and machine learning techniques," *Neural Computing and Applications*, vol. 28, no. 10, 2017.

[14] Ö. Yıldırım, P. Pławiak, R. S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," *Computers in Biology and Medicine*, vol. 102, pp. 411–420, 2018.

[15] M. K. Uçar, M. R. Bozkurt, C. Bilgin, and K. Polat, "Automatic sleep staging in obstructive sleep apnea patients using photoplethysmography, heart rate variability signal and machine learning techniques," *Neural Computing and Applications*, vol. 29, no. 8, 2018.

[16] P. Louridas and C. Ebert, "Machine learning," *IEEE Software*, vol. 33, no. 5, pp. 110–115, Sep. 2016.

[17] R. Rojas, "AdaBoost and the Super Bowl of Classifiers a Tutorial Introduction to Adaptive Boosting," *Computer Science Department*, Freie Universitat, Berlin, Germany, 2020, http://www.inf.fu-berlin.de/inst/ag-ki/adaboost4.pdf (2009).

[18] M. G. Pecht and M. Kang, "Machine Learning: Fundamentals," in *Proceedings of the Prognostics And Health Management Of Electronics: Fundamentals, Machine Learning, and the Internet Of Things*, p. 1, John Wiley & Sons, Hoboken, NJ, USA, 2019.

[19] N. Daldal, M. Nour, and K. Polat, "A novel demodulation structure for quadrate modulation signals using the segmentary neural network modelling," *Applied Acoustics*, vol. 164, p. 107251, 2020.

[20] M. Arican and K. Polat, "Binary particle swarm optimization (BPSO) based channel selection in the EEG signals and its application to speller systems," *Journal of Artificial Intelligence and Systems*, vol. 2, no. 1, pp. 27–37, 2020.

[21] G. Shobha and S. Rangaswamy, "Machine learning," *Handbook of Statistics*, vol. 38, pp. 197–228, 2018.

[22] N. Daldal, K. Polat, and Y. Guo, "Classification of multi-carrier digital modulation signals using NCM clustering based feature-weighting method," *Computers in Industry*, vol. 109, pp. 45–58, 2019.

[23] M. Paluszek and S. Thomas, *MATLAB Machine Learning*, Apress, New York, NY, USA, 1st edition, 2017.

[24] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, *Image Inpainting: A Review," Neural Processing Letters*, Springer, Berlin, Germany, 2019.

[25] O. Elharrouss, N. Almaadeed, and S. Al-Maadeed, "Mhad: multi-human action dataset," *Advances in Intelligent Systems and Computing*, vol. 1041, pp. 333–341, 2020.

[26] N. Almaadeed, O. Elharrouss, S. Al-Maadeed, A. Bouridane, and A. Beghdadi, *A Novel Approach for Robust Multi Human Action Detection and Recognition Based on 3-Dimentional Convolutional Neural Networks*, Elsevier, London, UK, 2019.

[27] K. Polat, "The fault diagnosis based on deep long short-term memory model from the vibration signals in the computer numerical control machines," *Journal of the Institute of Electronics and Computer*, vol. 2, no. 1, pp. 72–92, 2020.

[28] A. Ozdemir and K. Polat, "Deep learning applications for hyperspectral imaging: a systematic review," *Journal of the Institute of Electronics and Computer*, vol. 2, no. 1, pp. 39–56, 2020.

[29] C. Aldrich and L. Auret, "Artificial neural networks," *In Unsupervised Process Monitoring And Fault Diagnosis With Machine Learning Methods*, pp. 71–115, Springer London, Berlin, Germany, 2013.

[30] N. Daldal, "A novel demodulation method for quadrate type modulations using a hybrid signal processing method," *Physica A: Statistical Mechanics and Its Applications*, vol. 540, p. 122836, 2020.

[31] N. Daldal, Z. Cömert, and K. Polat, "Automatic determination of digital modulation types with different noises using Convolutional Neural Network based on time-frequency information," *Applied Soft Computing*, vol. 86, p. 105834, 2020.

[32] K. Polat and K. Onur Koc, "Detection of skin diseases from dermoscopy image using the combination of convolutional neural network and one-versus-all," *Journal of Artificial Intelligence and Systems*, vol. 2, no. 1, pp. 80–97, 2020.

[33] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[34] C. Mazo, J. Bernal, M. Trujillo, and E. Alegre, "Transfer learning for classification of cardiovascular tissues in histological images," *Computer Methods and Programs in Biomedicine*, vol. 165, pp. 69–76, 2018.

[35] J. Huang, Z. L. Yu, and Z. Gu, "A clustering method based on extreme learning machine," *Neurocomputing*, vol. 277, pp. 108–119, 2018.

[36] Matlab, "Deep Learning Toolbox Model for AlexNet Network-File Exchange-MATLAB Central," Mathworks, 2018, https://www.mathworks.com/matlabcentral/fileexchange/59133-deep-learning-toolbox-model-for-alexnet-network.

[37] Matlab, "Classify Image Using GoogLeNet - MATLAB, Simulink," Mathworks, 2018, https://www.mathworks.com/help/deeplearning/examples/classify-image-using-googlenet.html.

[38] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017*, vol. 5, pp. 3470–3479, Sydney, Australia, August 2017.

[39] J. Senthilnath, N. Varia, A. Dokania, G. Anand, and J. A. Benediktsson, "Deep TEC: deep transfer learning with ensemble classifier for road extraction from UAV imagery," *Remote Sensing*, vol. 12, no. 2, p. 245, 2020.

[40] A. Zhang, H. Wang, S. Li et al., "Transfer learning with deep recurrent neural networks for remaining useful life estimation," *Applied Sciences*, vol. 8, no. 12, p. 2416, 2018.

[41] I. H. Witten, E. Frank, and M. A. Hall, *Credibility: Evaluating What's Been Learned," in Data Mining: Practical Machine Learning Tools And Techniques*, pp. 147–187, Morgan Kaufmann, Burlington, MA, USA, 2011.

[42] D. M. W. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, pp. 2229–3981, 2011.

[43] R. Alpar, *Applied Statistic and Validation - Reliability*, Detay Publishing, London, UK, 2010.

[44] R. Koch, *The 80/20 Principle and 92 Other Powerful Laws of Nature: The Science of Success*, John Murray Press, London, UK, 2014.

[45] G. C. Cawley and N. L. C. Talbot, "Gene selection in cancer classification using sparse logistic regression with Bayesian regularization," *Bioinformatics*, vol. 22, no. 19, pp. 2348–2355, Oct. 2006.

[46] D. O. Richard, H. E Peter, and S. G. David, *Pattern Classification*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2001.

[47] L. Yu and H. Liu, "Feature selection for high-dimensional data: a Fast correlation-based filter solution," in *Proceedings of the Twentieth International Conference On Machine Learning*, Washington, DC, USA, December 2003.

[48] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, New York, NY, USA, 1991.

[49] H. Hanchuan Peng, F. Fuhui Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[50] L. J. Wei, "Asymptotic conservativeness and efficiency of kruskal-wallis test for K dependent samples," *Journal of the American Statistical Association*, vol. 76, no. 376, pp. 1006–1009, 1981.

[51] C. Gini, *Variabilità e mutabilità*, Reprinted in Memorie di metodologica statistica, Rome, 1912.

[52] H. Liu and H. Motoda, *Computational Methods of Feature Selection*, Chapman & Hall/CRC, Boca Raton, FL, USA, 2007.

[53] G. C. Cawley, N. L. Talbot, and M. Girolami, "Sparse multinomial logistic regression via bayesian L1 regularisation," in *In Advances In Neural Information Processing Systems 19*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds., pp. 209–216, MIT Press, London, UK, 2007.

[54] B. Junior and M. do C. Nicoletti, "An iterative boosting-based ensemble for streaming data classification," *Inf. Fusion*, vol. 45, pp. 66–78, Jan. 2019.

[55] T. Yamane, *Elementary Sampling Theory*, Prentice-Hall, Upper Saddle River, NJ, USA, 1st edition, 1967.

[56] E. B. Joaquin Quiñonero-candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, Eds., *Dataset Shift in Machine Learning (Neural Information Processing)*, MIT Press, London, UK, 2009.

[57] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: domain adaptation using asymmetric kernel transforms," in *Proceedings Of the IEEE Computer Society Conference On Computer Vision And Pattern Recognition*, pp. 1785–1792, Providence, RI, USA, June 2011.

[58] T. Lang and P. Ducimetière, "Newsweeder: Premature cardiovascular mortality in France: divergent evolution between social categories from 1970 to 1990," *International Journal of Epidemiology*, vol. 24, no. 2, pp. 331–339, 1995.

[59] J. Li, K. Cheng, and S. Wang, "Feature Selection: A Data Perspective," 2016, http://arxiv.org/abs/1601.07996.

[60] J. Li, K. Cheng, and S. Wang, "Feature Selection: A Data Perspective," 2018, http://featureselection.asu.edu/.

[61] K. Polat and S. Güneş, "A new feature selection method on classification of medical datasets: kernel F-score feature selection," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10367–10373, 2009.