ICONDATA

INTERNATIONAL CONFERENCE ON DATA SCIENCE & APPLICATIONS

PROCEEDINGS BOOK

VOLUME 1
FULL-TEXT BOOK

June 4-6, 2021

EDITOR
Dr. Murat GÖK

www.icondata.org

The Effect of Recursive Feature Elimination with Cross-Validation Method on Classification Performance with Different Sizes of Datasets

Berke AKKAYA[1]*

[1]*Istanbul University, School of Business, Quantitative Methods Department, Istanbul, TURKEY*

Abstract

The high dimensionality problem, which is one of the problems encountered in classification problems, arises when there are too many features in the dataset. This problem affects the success of classification models and causes loss of time. Feature selection is one of the methods used to eliminate the high dimensionality problem. Feature selection is defined as the selection of the best features that can represent the original dataset. This process aims to reduce the size of the data by reducing the number of features in the dataset by selecting the most useful and important features for the relevant problem. In this study, the performances of various classification algorithms in different data sizes were compared by using the recursive feature elimination method with cross-validation, which is one of the feature selection methods. Recursive feature elimination with cross-validation is a method that tries to get the most accurate result by eliminating the least important variables with cross-validation. In the study, datasets containing binary classification problems with a balanced distribution were used. Accuracy, ROC-AUC score, and fit time were used as evaluation metrics, while Logistic Regression, Support Vector Machines, Naive Bayes, k-Nearest Neighbors, Stochastic Gradient Descent, Decision Tree, AdaBoost, Multilayer Perceptron, and XGBoost classifiers were used as classification algorithms in the study. When the findings obtained as a result of recursive feature elimination with cross-validation were examined, it was observed that the accuracy increased by 5% on average and the ROC-AUC score increased by 5,3% on average, and the fit time decreased by about 5,1 seconds on average. It has been concluded that Naive Bayes and Multilayer Perceptron classifiers are the most sensitive to feature selection since they are the classifiers whose classification performance increases the most after feature selection.

Keywords: *Classification, Feature Selection, Machine Learning*

1 Introduction

The rapid development of information technologies and the cheaper use of these technologies have made large databases more accessible in recent years. As a result, the amount of data stored in databases today is increasing exponentially. In analyzing the data in these databases, modern methods such as data mining and machine learning have started to be used instead of traditional methods. These up-to-date methods can be defined as the set of processes that are useful to discover hidden information and various patterns in large data heaps [1]. Feature selection is part of this whole process.

Although every variable is seen to be effective in solving the problem in theory, there may be unnecessary, irrelevant, noisy, and misleading variables in the data used in solving many real-life problems, and there is usually no preliminary information about the models to be established to solve these problems. In cases where the number of independent variables is too high, the number of alternative variables that can be selected for the model also increases. In such cases, trying sub-sets of all possible variables and deciding in this way is not realistic both in terms of the time and cost of the application and the performance of the algorithm to be used in the analysis [2]. In the light of all this information, feature selection has become a very important issue in multidimensional data used

*e-posta: berkeakkaya@istanbul.edu.tr

today before the application of analysis and model building processes.

2 Feature selection

In current approaches, each variable in the dataset is called a feature or attribute. The reason why each of these variables is called a feature is that it reflects the characteristics of the sample represented by the dataset. Feature selection is defined as choosing the best subset of features that can represent the original dataset. Feature selection is the process of selecting the best k features among n features in the dataset by evaluating the features according to the algorithm used [3]. Feature selection aims to reduce the number of features in the dataset by selecting the most effective features for the problem of interest. When the number of features is not reduced, unnecessary features in the dataset may cause the training time of the model to increase and its interpretability to decrease. When the feature selection is applied:

- The working speed of the model increases.
- The understandability, interpretability, and visualization of the model become easier [4].
- It saves various costs such as time and storage.
- It reduces the error rate of the model [5].

Feature selection is usually a process that consists of the following steps [6].

1. *Generation of Subsets from the Original Dataset:* Subset generation, which is the process of obtaining smaller-sized datasets by using various strategies on the original dataset, is essentially a heuristic search process that determines candidate subsets in the search space to evaluate. This process consists of two basic stages. The first stage where the starting point/points of the search are decided, affects the direction of the search, this process can be forward, backward, or bidirectional. The second step is to decide which search strategy to use. For a dataset with M features, there are 2^M candidate subsets. As the number of features increases, the number of this search space also increases, and it becomes difficult to perform the search process. Therefore, different strategies such as sequential search and random search that can be used in various scenarios have been developed to overcome this difficulty [7, 8].
2. *Evaluation of Subsets:* In the evaluation stage, each new subset created in the first stage is evaluated with a specific evaluation criterion, and the optimal feature subset is tried to be found. A subset that is optimal for certain assessment methods may not be optimal by another criterion. Various methods are used to measure the optimality of the candidate subset. The first method of measuring this optimality is independent methods that do not need to work with any learning algorithm and allow calculating measures such as distance and information. Another method of measuring this optimality is dependent methods that allow calculating measures as a result of the output of various learning algorithms [7, 8].
3. *Stopping Criterion for Subset Selection:* Stopping criterion is the stage where it is determined when or in which situations the feature selection process should be stopped. Completion of the feature set search, reaching predetermined criteria, not finding a more optimal feature set are commonly known stopping criteria [2, 9].
4. *Results Validation of Selected Subsets:* While it is quite easy to verify the results in data with a priori knowledge, it is very difficult to have a priori knowledge in many applications where big data are used. For this reason, performance indicators of algorithms are preferred when feature selection is required while working with big data. Values such as error rate and accuracy are frequently chosen as performance indicators [10].

To summarize all these feature selection steps; first, a feature subset is created from the original dataset, and then an evaluation is made on the features using various methods and it is decided whether the relevant feature will be selected or not. The feature decided to be selected is included in the relevant subset and the process continues until the stop criterion is met [11].

2.1 Feature selection methods

The methods used in feature selection are generally divided into three groups: filter methods based on statistical information, wrapper methods based on performing search operations on features, and

embedded methods based on finding the best features through an algorithm. In filter methods, feature selection is made before running the algorithm to be used in the machine learning model, while in wrapper methods, the feature selection algorithm is used as a tool to select the best features. In embedded methods, the algorithm to be used in the machine learning model and the algorithm that will select the feature work together [12].

Although feature selection methods are used with all kinds of data with large dimensions; It is widely used in the process of determining important features related to the problem addressed in many areas such as text mining, diagnosis of diseases, fraud detection, churn analysis.

2.1.1 Filter Methods

As can be seen from the name, these methods enable selection by filtering the subset of the features in the dataset. In filter methods, the machine learning model is created after the features are selected. It is the oldest feature selection method since various statistical criteria are used. In these methods, the best feature subset is selected by calculating the values such as distance, information gain, and correlation of the samples in the dataset without using any algorithm [13]. The reason why the method is called filtering is the selection of feature subsets with the most appropriate values among the calculated statistical criteria. In a way, this method is the process of filtering features according to the statistical values obtained.

Filtering methods determine the various relationships (such as distance, information gain) of input variables with the target variable using statistical methods and then, select the variable with the strongest relationships. In other words, these methods make feature selection by allowing to see how much the change in this selected feature would change the target variable if only a certain feature was selected among all features [14]. When the literature and real-life applications are examined, it is seen that many different filter methods are used. Fisher Score, Chi-Square Test, Information Gain, and Correlation Criteria are commonly used filter methods.

Fisher Score Method calculates a score for each class value using the mean and standard deviation of the features.

$$F(w) = \frac{(\mu_1 - \mu_2)}{(\sigma_1^2 - \sigma_2^2)} \qquad (1)$$

In the formula of the fisher score [15] method mentioned in (1), values 1 and 2 represent two different classes, μ denotes the arithmetic mean of classes, and σ^2 denotes the standard deviations of the classes. In this method, features with high scores in data instances in the same class are assigned to similar values, while features in data instances in different classes are assigned different values. Thus, the features that will distinguish the classes from each other in the best way are selected. With this method, which tries to find the features that help to distinguish the two classes from each other, the feature selection process is made by selecting the desired number of features starting from the highest value, after listing the features according to the calculated scores [16, 17]. If the Fisher Score is high, it can be interpreted that the mean difference between the two classes for the relevant feature is wide and there are also small deviations in the relevant classes.

Chi-Square Test, which is based on whether the difference between observed and expected frequencies is significant, is one of the frequently used feature selection methods. The method tests whether there is a relationship between the input variables, which are features, and the target variable. Features found to be unrelated to the target variable as a result of the applied test are removed from the dataset [18]. The formula used for the Chi-Square test are given in (2).

$$\chi^2 = \sum_{i=1}^{i} \sum_{j=1}^{j} \frac{(f_{ij} - f'_{ij})^2}{f'_{ij}} \qquad (2)$$

In the formula [19] while the f_{ij} value denotes observed and the f'_{ij} value denotes the expected value, the degree of freedom of the chi-square distribution to be used for test statistics is calculated with the *d* value. It is calculated as *d = (i-1) * (j-1)*. In the Chi-Square method, when a certain number of features is desired to be selected, the process is done by selecting the desired number of features, starting from the highest value according to the calculated value of the features.

Information Gain Method works by calculating the difference between the information required to explain a selected feature and the information required to explain this selected feature together with any other feature. The entropy model is used to calculate this difference with the Information

Gain Score [20]. Entropy is expressed as a measure of uncertainty in a system.

$$H(Y) = -\sum_{Y} p(y)\log_2(p(y)) \quad (3)$$

$$H(Y|X) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x)\log_2(p(y|x)) \quad (4)$$

$$\text{Information Gain} = H(Y) - H(Y|X) \quad (5)$$

The Information Gain, which shows the decrease in the entropy value of the Y feature depending on the X feature [21], is calculated as in (3), (4), (5).

Correlation Criteria performs the feature selection process by using the correlation coefficient value. The correlation coefficient mathematically takes values between -1 and +1 and the strength of the relationship is expressed with these values. Using this criterion, the existence and strength of the relationship between the features and the target variable are investigated. This method is based on the hypothesis that the good feature subsets (input variables) consist of features with high correlation with the class (target variable) to be predicted and low correlation between each other [22, 23]. This means that features that have high correlations with each other are eliminated by this method and feature selection is made. In this method, the Pearson Correlation Value is frequently used.

$$R(i) = \frac{cov(x_i, Y)}{\sqrt{var(x_i) * var(y)}} \quad (6)$$

In equation (3), x_i is the i_{th} variable, Y is the target variable (class), $cov()$ is the covariance, and $var()$ is the variance. Since the correlation method can only detect linear dependencies between the features and the target variable, it is more accurate to use it with numerical feature subsets.

2.1.2 Embedded Methods

Embedded Methods perform feature selection by embedding feature selection with its classifier structure. This method involves the interaction of classification algorithms with feature selection algorithms. As with Wrapper Methods, it has the ability to work with classification algorithms that cannot be used in Filter Methods, although it has higher computational costs [11]. Decision Trees are widely used for classification problems in embedded feature selection [2], while Regularization Methods are commonly used for regression-based problems in embedded feature selection.

Decision Trees Method works by pruning the branches of the tree according to the importance of the input variables (features) of the classification algorithm. Any decision tree algorithm used for classification problems can be preferred to be used in this method. Like the Wrapper Methods, in the Decision Trees Method, feature set selection is made until the model reaches the best point. The importance of the features is revealed with various criteria in Decision Trees. After determining the importance limit, the features below this limit are not included in the model and the success performance of the model is increased [24].

Regularization Methods provide feature selection by penalizing input variables (features). The method determines the features that will have the most optimum effect on the target variable to achieve maximum accuracy while decreasing or increasing the penalty coefficients of these features [25]. There are 3 types of Regularization Methods: LASSO (L1), RIGDE (L2), and Elastic Nets (L1/L2) [11].

2.1.3 Wrapper Methods

Wrapper methods allow the selection of features that can show the best prediction performance with various learning algorithms for feature selection. Although these methods give better results than filtering methods in determining the best sub-feature set, they have higher computational costs [2]. Sequential Forward Selection, Sequential Backward Selection, Plus L - Minus R, Sequential Forward Floating Selection, Sequential Backward Floating Selection, and Recursive Feature Elimination methods are the commonly used wrapper methods.

Sequential Forward Selection (SFS) Method starts from an empty feature set and aims to find the subset with the best features by adding a feature that has not been added to the feature set before in each iteration. The contribution of the relevant feature to the success of the classification is considered when deciding whether to include any feature in the subset. The method continues until there is no increase in classification success or until stopping criteria are met. The feature selected in any repetition cannot be removed from the subset afterwards because the classification success increases even once.

Sequential Backward Selection (SBS) Method aims to find the subset with the best features to maximize

the classification success by reducing the number of features among all available features. This method starts the operation with all features, after that it continues by removing one feature from the subset at each step until there is no increase in the classification success, unlike the Sequential Forward Selection method. Once the features are removed from the subset, they cannot be included in the subset again in the Sequential Backward Selection method.

Plus L – Minus R Method can be called a combined version of the SFS and SBS method. This method first adds *L* number of features to the subset at each step. Then, it removes the number of *R* features from the subset to ensure that the classification success is optimum. *L* and *R* are constant values given during application. For these L and R values, any optimum value is not selected or created during the application. Therefore, it becomes difficult to reach the most optimum result unless the correct values are given. Plus L – Minus R Method can also be used to eliminate the problems such as not removing the selected feature from the feature set in the SFS method or not including a feature extracted in the feature set in the SBS method.

Sequential Forward Floating Selection (SFFS) Method is used as an alternative to the Plus L – Minus R method. Since the L and R values in the Plus L – Minus R are chosen randomly, the result obtained depends on certain L and R values. For this reason, the results obtained with the Plus L - Minus R method may not be optimum. In order to overcome this problem, a moving structure is proposed instead of fixing L and R values as in the Sequential Forward Floating Selection method. Thus, at any step of the feature selection in this method, the current classification success is moved in the same direction until it reaches a higher value. This method performs the feature selection process starting from an empty set.

Sequential Backward Floating Selection (SBFS) Method works in the opposite way to the SFFS method. This method performs the feature selection process starting with the whole set, in other words with all the features.

2.2 Recursive Feature Elimination with Cross-Validation

Recursive Feature Elimination, which is one of the wrapper methods, aims to reach the optimal number of features by eliminating the least important features. This method performs the feature selection process by working with a machine learning algorithm according to the success performance of the algorithm used. The method removes unnecessary and weak features that least affect the success of the classification model and at the same time maintains effective and strong features that improve the success of the model. It uses an iterative procedure that works similar to the backward feature selection. This method first creates the model on the entire feature set and grades each feature according to its effect and importance on the target variable. After that, it rebuilds the model after removing the least important feature at each step and recalculates the importance of each feature until the highest model success is achieved [26].

Recursive Feature Elimination with Cross-Validation (RFECV) method works just like Recursive Feature Elimination. The only difference is that it performs feature selection with cross-validation. It uses K-fold as a cross-validation technique. In the training of a classification model, the dataset is divided into two as training and testing. While performing this division, some deviations (bias) and errors may occur in the training and testing of the model depending on the distribution of the data. K-fold cross-validation, which can be used in such situations, divides the data into equal parts according to a specified number of k, allowing each part to be used for both training and testing. Thus, it minimizes deviations and errors caused by distribution and division. The RFECV method applies the same process procedure as in Recursive Feature Elimination to each of the k divided datasets. Although the RFECV method causes an additional data processing cost, it provides more reliable results [27], [28].

3 Material & methods

The Recursive Feature Elimination with Cross-Validation (RFECV) Method is used for feature selection in this study, while the k-fold cross-validation value in the RFECV, namely k value, was determined as 10.

In order to see the effect of RFECV on different sized datasets, 5 different data sets were used in the study. ARCENE is the first of these datasets. ARCENE aims to predict cancer cells by binary classification methods. While there are 10.000 features in the ARCENE dataset, the number of instances is 100. The second dataset used in the study is GISETTE. This dataset aims to predict

handwritten single-digit numbers using the classification methods. There are 5.000 features and 6.000 instances in the GISETTE dataset. MADELON is another dataset used in the study. This dataset with 500 features and 2.600 samples reflects a binary classification problem with continuous features. ARCENE, GISETTE, and MADELON datasets were used in the NIPS (Neural Information Processing Systems Conference) - Feature Selection Competition held in 2003 [29, 30]. The fourth dataset used in the study is HILL-VALLEY with 100 features and approximately 1200 instances. The HILL-VALLEY dataset reflects the problem of binary classification with coordinates representing a hill or valley, aiming to predict whether these coordinates belong to a hill or a valley [31]. The last dataset used in the study is SPECTF. This dataset aims to diagnose tomography results using the binary classification method. SPECTF has 45 features and about 350 instances [32].

In the classification process, firstly, each dataset is divided into two separate groups as 70% train and 30% test. Afterwards, the model was established on the training dataset and the classification models were evaluated by applying the established model to the test dataset. For the classification model, 9 different classification algorithms, namely Logistic Regression (LR), Support Vector Machines (SVM), Naive Bayes (NB), k-Nearest Neighbors (k-NN), Stochastic Gradient Descent (SGD), Decision Tree (DT), AdaBoost (ADA), Multi-layer Perceptron (MLP) and XGBoost (XGB), were used to see how different algorithms were affected by the feature selection in different sizes of datasets. In order to see the effect of the RFECV method; Accuracy and ROC-AUC values used in the evaluation of the classification algorithms and the fit time of the model were examined. A brief summary of the working principles of classification algorithms used in the study is given in the next paragraph.

The logistic regression algorithm performs classification using a logistic function with probabilities describing the possible results of a single experiment [33]. The support vector machine algorithm is the representation of train data divided into classes as points in space at the width it allows. In this way, classification is carried out according to the classes represented in space [34]. The Naive Bayes algorithm is based on the Bayesian theorem, with the assumption of independence between each pair of features [35]. k-Nearest Neighbors is a method of classification based on neighbors. By working with distance calculations, it classifies each data point by assigning the nearest neighbors specified [36]. Stochastic Gradient Descent is a simple and effective approach compatible with linear models. This algorithm, which can be useful when the dataset is large, supports different loss functions and penalty values for classification [37]. Decision Tree, as its name suggests, is a tree-based algorithm. It generates and classifies an array of rules using attributes in the data [38]. AdaBoost aims to achieve a powerful classifier that provides high accuracy by combining multiple classifiers that perform poorly. The main principle of AdaBoost is to train the data sample by adjusting the weights of the classifiers in each iteration [39]. Multilayer Perceptron is an artificial neural network with backward propagation. It consists of input layers, hidden layers, and an output layer. Layers use a non-linear activation function, except for the input layer. The activation function provides a curvilinear match between input and output. This algorithm allows classification by changing the weights of the neurons it assigns to it [40]. XGBoost can be described as a more regular form of the Gradient Boosting algorithm. XGBoost algorithm uses advanced regularization, which improves model generalization capabilities. It is simply a classification method that makes estimates by combining outputs from trees. Its main principle is to create new trees, reducing the errors of previous trees [41].

Finally, the Wilcoxon Signed-Rank Test was applied to compare the classification evaluation metrics obtained before and after the feature selection process with RFECV and to test whether these metrics show a statistically significant difference. The significance level was taken as 0.05 in all tests applied. While all feature selection and classification process in the study was implemented in Python 3.7.1, Wilcoxon Signed-Rank Test was performed in IBM SPSS 23.

4 Results

The optimum feature number of the ARCENE dataset was found to be 4.522 with the application of RFECV (k = 10). This optimum feature number is less than half of the original feature number (10.000) in the dataset. It is seen in Table 1 that the RFECV applied to the ARCENE dataset increases the Accuracy of the classification model and the ROC-AUC score used as a criterion to distinguish the predicted classes in the model for all algorithms

used in the study. Thus, when Table 1 is examined, it is seen that the algorithms most affected by the feature selection process are Naive Bayes and Multi-layer Perceptron. In the Naive Bayes algorithm, the accuracy of the classification model increased by 16% after feature selection, while the ROC-AUC score used as a criterion to distinguish the predicted classes in the model increased by 18%. In the Multilayer Perceptron algorithm, the accuracy of the classification model increased by 21% after the feature selection, while the ROC-AUC score used as the criterion to distinguish the predicted classes in the model increased by 12%. However, the Fit time of the classification process is significantly shortened for AdaBoost, Multilayer Perceptron, and XGBoost algorithms.

Table 1. Comparison of Classification Evaluation Metrics Before and After Feature Selection for ARCENE Dataset

	Accuracy (%)		**ROC-AUC (%)**		**Fit Time (sec)**	
	Orig.	*Feat. Sel.*	*Orig.*	*Feat. Sel.*	*Orig.*	*Feat. Sel.*
LR	95	95	94	99.7	0.95	0.13
SVM	92	97	95	99.8	0.55	0.31
NB	54	70	53	71	0.11	0.05
KNN	83	87	90	92	0.25	0.18
SGD	80	90	92	99	0.13	0.04
DT	75	76	75	76	0.43	0.25
ADA	76	78	83	88	5.81	2.55
MLP	74	95	88	99.4	4.21	1.68
XGB	83	85	91	94	4.30	1.95

In the GISETTE dataset, the optimum feature number was found to be 1.327, and this optimum feature number value is considerably lower than the original feature number in the GISETTE dataset. When Table 2 is examined, it is seen that the RFECV process increases the accuracy and ROC-AUC values in this dataset. In the GISETTE dataset, the algorithm most affected by the RFECV method is Naive Bayes with 23% accuracy and ROC-AUC score change. When the fit times of the algorithms in the GISETTE dataset before and after the feature selection in the last column of Table 2 are examined, it is seen that there is a significant fit time improvement in the Support Vector Machines algorithm.

Table 2. Comparison of Classification Evaluation Metrics Before and After Feature Selection for GISETTE Dataset

	Accuracy (%)		**ROC-AUC (%)**		**Fit Time (sec)**	
	Orig.	*Feat. Sel.*	*Orig.*	*Feat. Sel.*	*Orig.*	*Feat. Sel.*
LR	97	99	99	99	6.4	1.5
SVM	97	99	99	99	80.6	8.2
NB	71	94	73	96	2.0	0.3
KNN	95	96	98	99	10.7	2.6
SGD	96	99	99	99	1.3	0.3
DT	93	93	93	93	12.0	3.3
ADA	95	95	98	98	35.5	10.0
MLP	97	99	97	97	36.3	11.6
XGB	98	98	98	98	40.5	11.1

In the MADELON dataset, the number of features has dropped by a large margin from 500 to 10. Table 3 shows the Accuracy, ROC-AUC, and Fit Time values before and after feature selection of the MADELON dataset. When classifying with the original feature number of 500 in this data set, the accuracy, and ROC-AUC score are not at a satisfactory level in other algorithms except the XGBoost algorithm. On the contrary, when the classification process was performed with the optimal number of features found as 10 with the RFECV method, the accuracy rate increased by 36% in the k-Nearest Neighbors algorithm, while the ROC-AUC score increased by 39%. At the same time, the Accuracy increased by 30% in the Multi-layer Perceptron algorithm, while the ROC-AUC score increased by 33%. This is an indication that significant improvements have been made in the k-Nearest Neighbors and Multi-layer Perceptron algorithm when the feature selection is made with RFECV. Looking at the fit times of the algorithms, a great time saving was achieved with the feature selection in Support Vector Machines, but the fit time in the Multi-layer Perceptron algorithm increased after the feature selection.

Table 3. Comparison of Classification Evaluation Metrics Before and After Feature Selection for MADELON Dataset

	Accuracy (%)		**ROC-AUC (%)**		**Fit Time (sec)**	
	Orig.	*Feat. Sel.*	*Orig.*	*Feat. Sel.*	*Orig.*	*Feat. Sel.*
LR	56	61	58	64	0.16	0.05
SVM	56	62	58	64	46.80	0.28
NB	59	59	64	63	0.04	0.01
KNN	54	90	56	95	0.23	0.01

SGD	55	59	57	61	0.40	0.01
DT	75	82	75	82	0.77	0.01
ADA	61	60	66	64	3.50	0.25
MLP	58	88	61	94	5.15	17.20
XGB	81	87	89	93	2.68	0.27

The number of features in the HILL-VALLEY dataset has almost halved after RFECV processing, from 100 to 47. Table 4 shows the Accuracy, ROC-AUC, and Fit Time values before and after feature selection of the HILL-VALLEY dataset. In this dataset, when the classification is made with the original number of features, the accuracy rates are not at a satisfactory level. However, when the ROC-AUC scores are examined, only the ROC-AUC scores of the Logistic Regression and Support Vector Machine algorithms are at a satisfactory level. Low accuracy and high ROC-AUC means that algorithms are good at discriminating classes but are not good at predicting. However, when the classification process is performed with the optimal number of features found as 47 with the RFECV method, a satisfactory result cannot be obtained in terms of Accuracy. The highest increase in accuracy rate after feature selection was seen in the Multilayer Perceptron algorithm with 6%, while the highest ROC-AUC score increase rate after feature selection was seen in the Stochastic Gradient Descent algorithm with 7%. Looking at the fit times of the algorithms, a time saving was achieved with the feature selection in the AdaBoost algorithm, but in the Multi-layer Perceptron algorithm, the fit time increased slightly after the feature selection.

Table 4. Comparison of Classification Evaluation Metrics Before and After Feature Selection for HILL-VALLEY Dataset

	Accuracy (%)		**ROC-AUC (%)**		**Fit Time (sec)**	
	Orig.	*Feat. Sel.*	*Orig.*	*Feat. Sel.*	*Orig.*	*Feat. Sel.*
LR	70	70	95	96	0.20	0.08
SVM	61	61	93	94	0.28	0.09
NB	51	51	54	54	0.01	0.01
KNN	54	55	55	56	0.02	0.01
SGD	60	60	78	85	0.02	0.01
DT	56	54	56	54	0.29	0.12
ADA	52	53	54	54	0.77	0.53
MLP	61	67	71	76	2.97	3.76
XGB	55	56	57	60	0.79	0.42

The number of features, which was 45 in the original version of the SPECTF dataset, was found to be 3 as optimum after the RFECV process. This optimum feature number is considerably less than the number of original features that were already few in the data set. In this case, it can be decided whether the feature selection process should be done or not by the performance of the classifier. Table 5 shows that when RFECV is applied to SPECTF dataset, it increases Accuracy and ROC-AUC score in all algorithms except AdaBoost and XGBoost algorithms. Looking deeper into Table 5, which includes the evaluation criteria before and after the feature selection process, Naive Bayes showed the highest improvement with 11% in terms of Accuracy, while the k-Nearest Neighbors algorithm showed the highest improvement with 11% in terms of ROC-AUC Score. However, when looking at the AdaBoost algorithm, it is seen that the accuracy value has decreased. This shows that applying the feature selection process with RFECV to the SPECTF dataset adversely affects the prediction performance of the AdaBoost classifier. In such cases, a different feature selection method can be used, or the feature selection process may not be applied to datasets.

Table 5. Comparison of Classification Evaluation Metrics Before and After Feature Selection for SPECTF Dataset

	Accuracy (%)		**ROC-AUC (%)**		**Fit Time (sec)**	
	Orig.	*Feat. Sel.*	*Orig.*	*Feat. Sel.*	*Orig.*	*Feat. Sel.*
LR	80	82	82	88	0.02	0.01
SVM	78	81	80	88	0.01	0.01
NB	67	79	84	87	0.01	0.01
KNN	73	80	73	83	0.01	0.01
SGD	77	78	81	87	0.01	0.01
DT	73	76	60	62	0.01	0.01
ADA	82	77	83	83	0.09	0.08
MLP	78	83	83	88	1.76	0.65
XGB	80	80	81	80	0.11	0.04

In Table 6, the mean values for Accuracy, ROC-AUC Score, and Fit Time obtained before and after the feature selection process of each classification algorithm were calculated in the context of all data sets. Then, by finding the difference between these mean values, it is aimed to examine the general effect of RFECV on the data sets used in the study. When the findings in Table 6 were examined, it is seen that the accuracy increased by an average of 4.9%, the ROC-AUC score increased by an average of 5.3% and the Fit Time decreased by an average of 5.1 seconds.

Table 6. Differences of Mean Values of Accuracy, ROC-AUC Score and Fit Times Before and After Feature Selection

Dataset	Accuracy Difference (%)	ROC-AUC Difference (%)	Fit Time Difference (sec)
ARCENE	6.7	6.3	1.07
GISETTE	3.7	2.8	19.59
MADELON	10.3	10.9	4.63
HILLVALLEY	0.7	1.7	0.04
SPECTF	3.1	4.4	0.13
MEAN	**4.9**	**5.2**	**5.09**

When the results in Table 6 are examined, it is revealed that the feature selection with RFECV mostly improves the classification performance in the context of the algorithms used in the study, but it is necessary to test whether this difference in each dataset is statistically significant. In order to apply this test procedure, the Accuracy, ROC-AUC Score, and Fit Time values specified separately for each data set were combined. Then, it was tested whether the differences of these metrics were significant before and after the feature selection process with RFECV. Since the obtained Accuracy, ROC-AUC Score, and Fit Time values did not show the normal distribution and were determined as before and after a feature selection process, the Wilcoxon Signed-Rank Test was used for group comparison. Table 7 shows the results of this test for datasets.

Table 7. The Results of Wilcoxon Signed-Rank Test for Datasets

	Accuracy		ROC-AUC		Fit Time	
	Pos. Rank	*p*	*Pos. Rank*	*p*	*Neg. Rank*	*p*
ARCENE	8	0.012*	9	0.008*	9	0.008*
GISETTE	6	0.026*	2	0.180	9	0.008*
MADELON	7	0.017*	7	0.021*	8	0.086
HILL VALLEY	4	0.336	6	0.089	7	0.161
SPECTF	7	0.079	7	0.017*	4	0.066

*Difference is significant at the 0.05 level.

When the Accuracy column in Table 7 is examined, it can be said that the difference of Accuracy value in the ARCENE (*p=0.012*), GISETTE (*p=0.026*), and MADELON (*p=0.017*) datasets are significant. When considering these significant results, it is seen that the Accuracy performance of 8 algorithms in the ARCENE dataset, 6 algorithms in the GISETTE dataset, and 7 algorithms in the MADELON dataset have increased positively when the RFECV method was applied. Considering the ROC-AUC column in Table 7, it can be said that the difference of ROC-AUC Scores in ARCENE (*p=0.008*), MADELON (*p=0.021*), and SPECTF (*p=0.017*) datasets are significant. When the change in these significant results is examined, it is seen that the ROC-AUC Scores of 9 algorithms in the ARCENE data set, 7 algorithms in the MADELON data set, and 7 algorithms in the SPECTF data set increased positively after feature selection with RFECV. Looking at the last column of Table 7, it can be said that the difference between Fit Times in ARCENE (*p=0.008*) and GISETTE (*p=0.008*) datasets is significant. In ARCENE and GISETTE datasets, it is seen that Fit Times in 9 algorithms increase negatively after feature selection with RFECV. In this case, it can be interpreted that the fit times decrease for each algorithm in these two datasets.

Finally, it was tested whether the results of classification success metrics of each algorithm before and after feature selection with RFECV show a statistical significance or not. Table 8 shows the results of the Wilcoxon Signed-Rank test for each algorithm.

Table 8. The Results of Wilcoxon Signed-Rank Test for Each Algorithm

	Accuracy		ROC-AUC		Fit Time	
	Pos. Rank	*p*	*Pos. Rank*	*p*	*Neg. Rank*	*p.*
LR	3	0,102	4	0,066	5	0,043*
SVM	4	0,068	4	0,068	4	0,068
NB	3	0,109	3	0,144	3	0,109
KNN	5	0,042*	5	0,042*	4	0,068
SGD	4	0,068	4	0,066	4	0,066
DT	3	0,273	3	0,357	4	0,068
ADA	2	0,854	1	0,655	5	0,043*
MLP	5	0,043*	4	0,066	2	0,686
XGB	3	0,109	3	0,141	5	0,043*

When the significant results in Table 8 are examined, it is seen that the k-Nearest Neighbors and Multi-layer Perceptron algorithm is positively affected by the feature selection in the context of Accuracy. Accuracy rate increased in all data sets after feature selection in both algorithms. Considering the significant results in the context of ROC-AUC Score, the ROC-AUC Score of all data sets increased in the k-Nearest Neighbors algorithm after feature selection. However, in the significant results of the algorithms in Fit Times, the Fit Times

of all datasets in Logistic Regression, AdaBoost and XGBoost algorithms are shortened after feature selection.

5 Conclusion

Feature selection is defined as choosing the best subset that can represent the original data set. This feature selection process aims to reduce the size of the variables in the dataset, namely the features, by selecting the most effective features for the relevant problem. Nowadays, feature dimensions in the stored data are at quite high levels. Analyzing such high feature-size data reveals serious problems in terms of both the time & cost of the application and the performance of the machine learning algorithm to be used. For this reason, feature selection has become an important issue while analyzing for high-dimensional data.

Many different methods are used in feature selection. These methods have various advantages and disadvantages. The first of these methods is the filter methods that are based on statistical theories and applied before analysis such as classification and clustering, the second of these methods is the wrapper methods that perform search operations on features and use the algorithms used in methods such as classification and clustering as a tool for selecting the best features, the last of these methods are embedded methods that are based on finding the best divisor criterion and work concurrently with algorithms used in methods such as classification and clustering.

In this study, feature selection was made by using one of the wrapper methods, the Recursive Feature Elimination with Cross-Validation method. In classification analysis, Logistic Regression, Support Vector Machines, Naive Bayes, k-Nearest Neighbors, Stochastic Gradient Descent, Decision Tree, AdaBoost and XGBoost algorithms applied on 5 different datasets including ARCENE, GISETTE, MADELON, HILL-VALLEY and SPECTF which contain binary classification problems with balanced distribution. Accuracy, ROC-AUC Score, and Fit Time values were used to evaluate the classification success of these algorithms. When these values obtained before and after feature selection with RFECV in each dataset were examined, it was seen that the accuracy increased by an average of 5%, the ROC-AUC score increased by an average of 5.3% and the Fit Time decreased on average by about 5.1 seconds. In general, it was concluded that Naive Bayes and Multilayer Perceptron classifiers are most sensitive to feature selection since they are the algorithms whose classification performance increases the most after feature selection. Finally, these differences occurring before and after feature selection with RFECV were statistically tested with Wilcoxon Signed-Rank Test. With this test, it was concluded that the increase in accuracy value after feature selection was significant in ARCENE, GISETTE, MADELON datasets, and the increase in ROC-AUC Score after feature selection was significant in ARCENE, MADELON, SPECTF datasets. However, the decrease in Fit Time in ARCENE and GISETTE datasets was found to be significant.

In conclusion, it has been observed that the classification results obtained when the feature selection is made with the Recursive Feature Elimination with Cross-Validation method, which is one of the wrapper feature selection methods, is more successful in datasets with higher feature sizes. Therefore, it can be said that the Recursive Feature Elimination with Cross-Validation method can be used in the feature selection process in datasets with high feature sizes and can give better results.

References

[1] Coenen F. "Data mining: past, present and future". Knowledge Engineering Review, 26(1), 25-29, 2011.

[2] Guyon I, Elisseeff A. "An Introduction to Variable and Feature Selection". Journal of Machine Learning Research, 3, 1157-1182, 2003.

[3] Kira K, Rendell L.A. "A Practical Approach to Feature Selection". Machine Learning Proceedings, 249-256, 1992.

[4] James G, Witten D, Hastie T, Tibshirani R. An Introduction to Statistical Learning. Springer, 204, 2013.

[5] Bermingham M. L, Pong-Wong R, Spiliopoulou A, Hayward C, Rudan I, Campbell H, Wright A. F, Wilson J. F, Agakov F, Navarro P, Haley C. S. "Application of high-dimensional feature selection: evaluation for genomic prediction in man". Scientific Reports, 5(10312), 1-12, 2015.

[6] Li J, Cheng K, Wang S, Morstatter F, Trevino R, Tang J, Liu H. "Feature Selection: A Data Perspective". ACM Computing Surveys, 50, 2016.

[7] Dash M, Liu H. "Feature Selection for Classification". Intelligent Data Analysis, 1, 131–156, 1997.

[8] Liu H, Yu L., "Towards Integrating Feature Selection Algorithms For Classification And Clustering". IEEE Transactions on Knowledge and Data Engineering, 17(4), 491-502, 2005.

[9] Saw T. "Swarm Intelligence Based Feature Selection for High Dimensional Classification: A Literature

Survey". International Journal of Computer, 33, 69-83, 2019.
[10] Sahu B, Dehuri S, Jagadev A. "A Study on the Relevance of Feature Selection Methods in Microarray Data". The Open Bioinformatics Journal, 11, 117-139, 2018.
[11] Tang J., Liu H, Alelyani S. "Feature Selection for Classification". Data Classification: Algorithms and Applications, 37, 37-64. Arizona, CRC Press, 2017.
[12] Saeys Y, Inza I, Larrañaga P. "A Review of Feature Selection Techniques in Bioinformatics". Bioinformatics, 23(19), 2507–2517, 2007.
[13] Sánchez-Marono N, Alonso-Betanzos A, Tombilla-Sanromán M. "Filter Methods for Feature Selection–a Comparative Study". International Conference on Intelligent Data Engineering and Automated Learning, Berlin, Heidelberg, 2007.
[14] Baraldi P, Zio E, Roverso D. "Feature Selection for Nuclear Transient Diagnostics", 2007. DOI: 10.13140/RG.2.1.2010.8321.
[15] Loo L.-H, Roberts S, Hrebien L, Kam M. "New Criteria for Selecting Differentially Expressed Genes". IEEE Engineering in Medicine and Biology Magazine, 26, 17-26, 2007.
[16] Aksu D, Ustebay S, Aydin M.A, Atmaca T., "Intrusion Detection with Comparative Analysis of Supervised Learning Techniques and Fisher Score Feature Selection Algorithm". Computer and Information Sciences: Communications in Computer and Information Science, Springer, 141-149, 2018.
[17] Gu Q, Zhenhui L, Jiawei H. "Generalized Fisher Score for Feature Selection" arXiv preprint arXiv:1202.3725, 2012.
[18] Jin X, Xu A, Bie R, Guo P. "Machine Learning Techniques and Chi-Square Feature Selection for Cancer Classification Using SAGE Gene Expression Profiles". BioDM 2006: Data Mining for Biomedical Applications, Berlin, Heidelberg, Springer, 106-115, 2006.
[19] Ozsoy O. İktisatçılar ve İşletmeciler için İstatistik. Turhan Kitapevi, Ankara, 269, 2002.
[20] Yang Y, Pedersen J.O. "A Comparative Study on Feature Selection in Text Categorization". 14th International Conference on Machine Learning, San Francisco, CA, USA, 1997.
[21] Taqwa A, Md Siraj M, Zainal A, Elshoush H.T, Elhaj F. "Feature Selection Using Information Gain for Improved Structural-Based Alert Correlation". PLOS ONE, 11, 2016.
[22] Hall M. A. Correlation-based Feature Selection, The University of Waikato, Hamilton: PhD Thesis, 1999.
[23] Chandrashekar G, Sahin F. "A Survey on Feature Selection Methods". Computers & Electrical Engineering, 40(19), 16-28, 2014.
[24] Ladha L, Deepa T. "Feature Selection Methods and Algorithms" International Journal on Computer Science and Engineering, 3(5), 1787-1797, 2011.
[25] Ma S, Huang J. "Penalized Feature Selection and Classification in Bioinformatics". Briefings in Bioinformatics, 9(5), 392-403, 2008.
[26] Guyon I, Weston J, Barnhill S, Vapnik V. "Gene Selection for Cancer Classification Using Support Vector Machines". Machine Learning, 46(1), 389-422, 2002.
[27] Mathew T. E. "A Logistic Regression with Recursive Feature Elimination Model for Breast Cancer Diagnosis". International Journal on Emerging Technologies, 10(3), 55-63, 2019.
[28] Misra P, Yadav A.S. "Improving the Classification Accuracy using Recursive Feature Elimination with Cross-Validation". International Journal on Emerging Technologies, 11(3), 659-665, 2020.
[29] NIPS, "NIPS 2003 Workshop on Feature Selection". British Columbia, CA, 2003.
[30] Guyon I, Gunn S, Nikravesh M, Zadeh L. Feature Extraction, Foundations and Applications. Berlin, Springer, 237-263, 2006.
[31] Graham L, Oppacher F. "Hill-Valley Dataset". http://archive.ics.uci.edu/ml/datasets/hill-valley.
[32] Cios K. J, Kurgan L. A. "SPECTF Dataset". http://archive.ics.uci.edu/ml/datasets/SPECTF+Heart.
[33] Worth A. P, Cronin M. T. "The Use of Discriminant Analysis, Logistic Regression and Classification Tree Analysis in the Development of Classification Models for Human Health Effects". Journal of Molecular Structure: THEOCHEM, 622(1-2), 97-111, 2003.
[34] Ghumbre S, Patil C, Ghatol A. "Heart Disease Diagnosis using Support Vector Machine". International Conference on Computer Science and Information Technology (ICCSIT'2011), Pattaya, 2011.
[35] Murphy K. P. "Naive bayes classifiers". University of British Columbia, 18(60), 2006.
[36] Liao Y, Vemuri V. R. "Use of k-Nearest Neighbors classifier for intrusion detection". Computers & Security, 21(5), 439-448, 2002.
[37] Bottou L., "Large-scale machine learning with stochastic gradient descent". Proceedings of COMPSTAT'2010, Berlin, Heidelberg, 2010.
[38] Safavian S. R, Landgrebe D. "A survey of decision tree classifier methodology". IEEE transactions on systems, man, and cybernetics, 21(3), 660-674, 1991.
[39] Hastie T, Rosset S, Zhu J, Zou, H. "Multi-class adaboost". Statistics and Its Interface, 2(3), 349-360, 2009.
[40] Öztemel E. Yapay Sinir Aglari. Papatya Yayincilik, Istanbul, 29-31, 2003.
[41] Chen T, Guestrin C. "XGBoost: A Scalable Tree Boosting System". 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, 2016.