Review Article GU J Sci 36(4): 1506-1520 (2023) DOI: 10.35378/gujs.993763

Gazi University

Journal of Science

http://dergipark.gov.tr/gujs

A Comprehensive Review of Feature Selection and Feature Selection Stability in Machine Learning

Mustafa BUYUKKECECI[1, *], Mehmet Cudi OKUR[2]

[1]*Univerlist, İzmir, Turkey*
[2]*Yasar University, Faculty of Engineering, Software Engineering Department, İzmir, Turkey*

Highlights

- This paper focuses on feature selection and feature selection stability.
- In the pages that follow, we review supervised feature selection methods.
- We examine several metrics used to measure the stability of the feature selection algorithms.

Article Info

Received: 10 Sep 2021
Accepted: 04 Sep 2022

Keywords

Feature selection, Dimensionality reduction, Types of feature selection, Feature selection stability, Stability measures

Abstract

Feature selection is a dimension reduction technique used to select features that are relevant to machine learning tasks. Reducing the dataset size by eliminating redundant and irrelevant features plays a pivotal role in increasing the performance of machine learning algorithms, speeding up the learning process, and building simple models. The apparent need for feature selection has aroused considerable interest amongst researchers and has caused feature selection to find a wide range of application domains including text mining, pattern recognition, cybersecurity, bioinformatics, and big data. As a result, over the years, a substantial amount of literature has been published on feature selection and a wide variety of feature selection methods have been proposed. The quality of feature selection algorithms is measured not only by evaluating the quality of the models built using the features they select, or by the clustering tendencies of the features they select, but also by their stability. Therefore, this study focused on feature selection and feature selection stability. In the pages that follow, general concepts and methods of feature selection, feature selection stability, stability measures, and reasons and solutions for instability are discussed.

1. INTRODUCTION

Datasets originating from different disciplines, such as biology, retail, finance, geography, and astrophysics, can contain thousands of features for every instance. Some of these features have little or no contribution (a.k.a., redundant and irrelevant features) to machine learning models. For this reason, eliminating them from the learning process helps to increase learning speed, reduce the risk of overfitting, overcome the curse of dimensionality[1], decrease computational requirements, create simple models with high generalization performance, and facilitate the interpretability of results. Selecting relevant features is an indispensable task, and feature selection approaches are used to detect such features.

The primary aim of feature selection is to find the optimal set of relevant features without losing the salient characteristics of the data. However, this is an NP-Hard problem [1, 2]. The relevancy of features can be classified as weak and strong [1]. Features with weak relevance mostly have a limited contribution to the machine learning process. Therefore, excluding these features from the model-building process does not always degrade the performance of machine learning algorithms. However, features with strong relevance always contribute to the machine learning process, and excluding them from the model-building process

[1] The curse of dimensionality means that machine learning algorithms suffer performance degradation as the number of features (dimensions) increases.

*Corresponding author, e-mail: mustafa.buyukkececi@univerlist.com

results in performance degradation. Yu and Liu [2] pointed out that "*an optimal feature subset should include all strongly relevant features, none of the irrelevant features, and a subset of weakly relevant features*". Feature selection has been used in a wide range of applications, such as DNA microarray analysis [3-6], hard drive failure prediction [7], intrusion detection [8, 9], image classification and retrieval [10, 11], text mining [12-14], etc.

Selection stability is an important property of feature selection algorithms. The stability of the feature selection algorithm is defined as the variation in the outputs of the selection algorithm due to slight differences in the training set (data). Unstable feature selection algorithms significantly change their feature preferences when a small amount of change is introduced to the training data. This makes it difficult for the user to identify feature subsets and undermines confidence in the algorithm and the overall analysis process. Like feature selection, feature selection stability also arouses the interest of researchers. Therefore, there is a considerable amount of published resources on this subject. This paper has been organized as follows. Section 2 gives a brief description of the feature selection process. Section 3 discusses types of feature selection, while Section 4 describes supervised feature selection methods. Section 5 reviews some common classification model evaluation metrics. The next two sections formulate the problem of feature selection stability and the metrics used to assess stability. Section 8 outlines reasons and solutions for the instability. Finally, Section 9 concludes the study.

2. FEATURE SELECTION

Datasets can contain irrelevant and redundant features[2] that adversely affect the machine learning process. Therefore, features that contribute to the machine learning process should be detected using feature selection techniques. Using domain knowledge, common sense, and getting help from domain expertise may also help to detect and eliminate redundant features. However, a more systematic approach is needed for high-dimensional datasets. Since redundant and irrelevant features have no significant impact on machine learning, discarding them from the learning process will help to increase learning speed, reduce overfitting, avoid the curse of dimensionality, and create simple models. For a dataset d, feature selection is to select a number of features, $f' = \{f_i' |\ i = 1,2,3, \dots, m\}$, from the original feature set $f\ = \{f_i |\ i = 1,2,3, \dots, n\}$ by satisfying conditions $m \subset n$ and $argmax_{f'}\ (T)$, i.e., maximizing the value of T, where T is a target function, such as classification accuracy or the quality of clusters.

The general procedure of supervised feature selection has four steps (see Figure 1). The first step is to explore the feature space using a search technique, e.g., random search, and to generate a candidate feature subset. The next step is to measure the goodness of the generated subset using a specified evaluation criterion, e.g., distance, mutual information, etc., or an inductive learner. The third step is to control the stopping criterion used to limit the run time of the selection process. Some common stopping criteria are: obtaining the optimal feature set, reaching the maximum number of iterations or a predefined number of features, or no improvement in the accuracy. If the stopping criterion is unsatisfied, the selection process returns to the first step. The last step is to validate the selected features. A classifier tries to predict the labels of the instances in the testing dataset on the selected features.

Unsupervised feature selection also, takes place in four steps: feature subset generation, feature subset evaluation, stopping criterion, and validation. In unsupervised feature selection, datasets are not split into training and test sets, and all samples are used in the selection process. As with supervised feature selection, feature selection can be independent or dependent on unsupervised learning algorithms. Iteration size or no improvement in the clustering quality[3] can be set as a stopping criterion. Finally, the results can be validated by conducting different experiments. Feature selection and feature extraction are two different approaches used for dimension reduction. Feature extraction, *or* feature construction, approaches construct new

[2] Irrelevant features provide no useful information to the machine learning process, and redundant features provide duplicate information.

[3] Clustering quality can be measured using an evaluation metric, such as Calinski-Harabasz Index (CH Index) or Silhouette Coefficient.

features using existing features without losing any information. LDA (Linear Discriminant Analysis) [15] is a well-known example of a feature extraction algorithm.

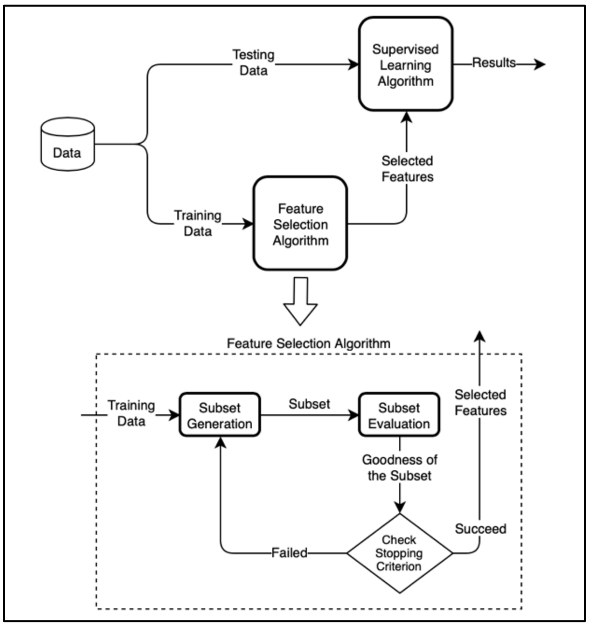

Figure 1. *Steps of supervised feature selection*

3. TYPES OF FEATURE SELECTION

Depending on the label availability, feature selection is classified as supervised, unsupervised, and semi-supervised. Most of the feature selection algorithms work with either labeled or unlabeled data. However, unified feature selection algorithms can work with both types of data [16, 17].

3.1. Supervised Feature Selection (*or* Feature Selection for Classification)

Supervised feature selection is performed on labeled data [18, 19]. Thus, the selection algorithms evaluate the relation between the features, i.e., independent variables, and the class variable, i.e., dependent variable, using an evaluation criterion (see Section 4.1) or a classifier. For supervised feature selection, irrelevant features are the ones with low or no association with the dependent variable. Supervised feature selection can be used in binary, e.g., the screening test for diseases, multi-class[4], e.g., credit rating classification, or multi-label[5], e.g., classification of movies, classification problems. The Chi-Square test and Neighborhood Component Analysis (NCA) [20] are examples of supervised feature selection algorithms.

3.2. Unsupervised Feature Selection (*or* Feature Selection for Clustering)

Manual data labeling is a costly and laborious task. Automated data labeling[6], on the other hand, still needs humans to check the accuracy of the labels. In addition to all, it is not always possible to classify instances. For this reason, unlabeled data are easily available and abundant. Unsupervised feature selection is performed on unlabeled data [19, 21, 22] and is a challenging task, as there are no class labels to guide the selection process. The basis for unsupervised feature selection is clustering tendency assessment, i.e., clusterability. Clustering tendency assessment is used to determine whether a given dataset contains meaningful, i.e., non-random, clusters. Datasets having an inherent grouping structure indeed have clusters and are suitable for clustering. Unsupervised feature selection algorithms select features that maximize this tendency. Therefore, for unsupervised feature selection, irrelevant features are the noisy ones that do not

[4] The term refers to “a classification problem with more than two classes”.

[5] The term refers to “a classification problem where each sample has more than one class label”.

[6] It is an automated data labeling approach that uses machine learning and artificial intelligence.

tend to form clusters. PCA [23] and Laplacian Score [24] are examples of unsupervised feature selection algorithms.

3.3. Semi-supervised Feature Selection

In semi-supervised feature selection, the relevance of the features is determined using both labeled and unlabeled data [19, 25-27]. Different strategies are used for semi-supervised feature selection. One is to construct a similarity matrix, i.e., affinity matrix, that uses labeled data to have distinctive information and unlabeled data to have complementary information. Another strategy is to use the manifold assumption [28]. Relatively little research has been done on semi-supervised feature selection, so this topic remains an open field of research. LSDF algorithm [29] is an example of a semi-supervised feature selection algorithm.

4. SUPERVISED FEATURE SELECTION METHODS

Several studies [18, 30, 31] have categorized supervised feature selection methods as filter, wrapper, and embedded. In this study, we classified the supervised feature selection methods into five main categories, namely, filter, wrapper, embedded, hybrid, and ensemble. Hybrid and ensemble methods use a combination of filters, wrappers, and embedded algorithms to make better selections and to avoid being dependent on the performance of a single feature selection algorithm or a result set. Both methods perform feature selection more than once and achieve selection diversity. For this reason, in this study, they were examined as separate categories. A list of some common feature selection algorithms is given in Table 1.

Table 1. *List of some common selection algorithms*

Method	Algorithms	Type/Search Strategy
Filter	mRMR ReliefF Chi-square Test	Univariate
	CFS (Correlation-Based Feature Selection) FCBF (Fast Correlation-Based Feature Selection) Markov Blanket-Based Filter Selection	Multivariate
Wrapper	SFS (Sequential Forward Selection) SBS (Sequential Backward Selection) Bidirectional Search	Deterministic Search
	Randomized Hill Climbing Genetic Search Simulated Annealing	Randomized Search
	Exhaustive Search Beam Search Branch and Bound	Exponential Search
Embedded	Decision Trees Random Forest Weighted Naïve Bayes	Tree-Based Methods
	Ridge Regression LASSO Elastic Net	Regularization Methods

4.1. Filter (*or* Feature Filtering) Methods

Filter methods do not select features, instead, they rank the entire feature set using an evaluation function, i.e., evaluation criteria. Feature selection is done by the user considering the rankings (relevancy scores). The evaluation function can be distance, information, i.e., entropy, accuracy, correlation, and consistency based. Filter methods use statistical and mathematical functions rather than classifiers to assess the goodness of features. Filter methods are categorized as univariate and multivariate. **Univariate filter methods** do not assess the relationships between the features (do not assess feature dependencies). This means the relevance of each feature in the feature set is evaluated independently, i.e., performs individual evaluation, which causes the features that are useless alone but valuable in combination with other features to be ignored. **Multivariate filter methods** assess the relationships between the features (assess feature dependencies) to some extent. Therefore, multivariate filter methods are slower and less scalable than univariate methods.

4.2. Wrapper Methods

Wrapper methods have three components: a search strategy, e.g., randomized search, to generate feature subsets, a classifier working as a black box[7] to assess the generated feature subsets, and a stopping criterion, e.g., reaching the maximum number of iterations. Wrappers consider dependencies among features and have better generalization ability than filters. For high-dimensional datasets, wrapper methods are computationally costly. This is because, for n number of features, it is possible to generate 2^n subsets and the classification performance of each subset must be validated by the classifier. Calling the classifier recurrently also makes wrapper methods prone to overfitting as compared to filter and embedded methods. Another disadvantage of this method is that the feature preferences of the wrapper methods depend on the classifier used as the black box.

4.3. Embedded Methods

Feature selection in embedded methods occurs whilst the classifier is being trained. Therefore, the feature selection process results in both a selected feature subset and a trained classifier. Feature preferences of embedded methods are affected by the classifier hypothesis. Embedded methods are computationally less intensive, less prone to overfitting, and have faster running time as compared with wrappers. They can also capture feature dependencies. The tree-based classification algorithms, such as CART [32] and ID3 [33] are examples of embedded methods.

4.4. Hybrid Methods

The general idea behind hybrid methods is to combine different feature selection approaches and leverage the strengths of selectors to achieve the best, i.e., optimal, results. For instance, a hybrid method can be constructed by combining filter and wrapper methods. The feature selection process works as follows. First, the entire feature set is ranked by a filter method. Then, the user generates a feature subset usually by heuristically setting a relevance threshold or by simply selecting the top n features. Finally, a wrapper method is employed to further reduce the generated feature subset. The main issue in hybrid methods is the successive use of different feature selection methods increases the computational cost.

4.5. Ensemble Methods

In ensemble methods, feature selection is performed more than once and the generated feature subsets are combined into a single subset. There are three different ensemble strategies. In the **data diversity strategy**, first, the original dataset is sampled using a sampling method, e.g., simple random sampling with/without replacement, and then a single selection algorithm is applied to each sample. In the **functional diversity strategy**, a set of different selection algorithms are applied to the original dataset without using any sampling method. In the **hybrid strategy**, different selection algorithms are applied to different datasets generated from the original dataset by sampling. Aggregating feature subsets have an important role in any ensemble scheme.

5. EVALUATING SUPERVISED MODELS

For supervised learning, feature selection quality is expressed by the classification performance, which is evaluated using various metrics, such as Accuracy Rate, Error Rate, Sensitivity, Specificity, AUC (Area under the Curve), etc. Table 2 summarizes some common evaluation metrics and their formulas. For a

[7] The term refers to “without considering the internal workings of the algorithm”.

binary classification problem, assume that class A is a positive class and class B is a negative class. True positives (TP) are the number of correctly labeled samples belonging to the positive class. True negatives (TN) are the number of correctly labeled samples belonging to the negative class. False positives (FP) are the number of incorrectly labeled samples belonging to the negative class. False negatives (FN) are the number of incorrectly labeled samples belonging to the positive class. See [34] for a detailed brief on model evaluation metrics.

Table 2. *Supervised model evaluation metrics and their formulas*

Metric	Formula
Accuracy Rate	$(TP + TN)/(TP + TN + FN + FP)$
Error Rate	$1 - Accuracy\ Rate$
Sensitivity	$TP/(TP + FN)$
Specificity	$TN/(TN + FP)$
Positive Prediction Value (PPV)	$TP/(TP + FP)$
Negative Prediction Value (NPV)	$TN/(TN + FN)$
Prevalence	$(TP + FN)/(TP + TN + FN + FP)$
F1 Score	$(2 * TP)/((2 * TP) + FP + FN)$
True Positive Rate (TPR)	$Sensitivity$
False Positive Rate (FPR)	$1 - Specificity$

The performance of binary classification models can be visualized using the ROC curve. The ROC curve is plotted in the two-dimensional ROC space, where TPR is on the vertical axis, and FPR is on the horizontal axis. Both axes are ranged between 0 and 1. Therefore, the ROC Curve goes through the points $(0,0)$ and $(1,1)$. To draw the ROC curve, the cut-off points, i.e., decision thresholds, are first determined by the user, and for each cut-off point, the TPR and FPR values are calculated. These values represent a point in the ROC space. When all cut-off points are over, the points marked on the ROC space are connected to form the ROC curve. The ROC curve is a graphical plot that portrays the performance of the binary classification models. For the numerical representation of the performance, the area under the curve (AUC) is calculated. The accuracy of the model is directly proportional to the area under the curve. That is, the larger the AUC value, the more accurate the model is. The ROC curve can be plotted for multiclass classification models using the “one versus one” and “one versus all” strategies [35, 36].

6. FEATURE SELECTION STABILITY

The stability of feature selection algorithms was first studied by Turney [37]. Slight changes in the training data may cause radical differences in the feature preferences of the selection algorithm. Stable feature selection algorithms are insensitive to these variations and do not change their feature preferences. On the other hand, unstable algorithms are sensitive to training data variations and change their feature preferences. Resampling the training set, removing or adding records to the training set, adding noise to records, and reordering the records or features are examples of data variation.

Stability is an important issue for feature selection algorithms because it is difficult to verify and interpret the results of an unstable algorithm. Factors such as unbalanced class distributions [38, 39], skewed data [40], outlier and noisy values in the dataset, features representing similar information and features that are closely correlated, i.e., multi-dependency and multicollinearity [41], insufficient number of samples, and high dimensionality, e.g., the curse of dimensionality [42], affect the stability of the selection algorithm. In addition to these, not choosing the right feature selection technique and incorrectly setting its hyperparameters also affect the stability.

Researchers have paid much attention to supervised feature selection and supervised feature selection stability. On the other hand, only a few studies have analyzed unsupervised and semi-supervised feature selection stability. The remaining part of the paragraph summarizes some studies related to these topics.

Wu and Chang [43] introduced a new unsupervised feature selection method that uses Neural Networks to score and select relevant features. The authors empirically analyzed the sensitivity of the selection algorithm using algorithmic stability analysis. Helleputte and Dupont [44] introduced a semi-supervised feature selection method based on regularized linear models. The authors used the Kuncheva's Index to measure the stability of the method and the Balanced Classification Rate to test the performance of the model built on selected features. Lai and Garibaldi [45] conducted experiments using four different feature selection methods and the Semi-supervised Fuzzy C-Means Classifier (as performance evaluator). The stability of the selection methods was quantified using the stability measure proposed by Kalousis et al. [46].

7. STABILITY MEASURES

Feature selection algorithms can represent their results, i.e., feature preferences, in terms of rank, weight, and index. Therefore, depending on the representation, stability measures can be classified as stability by rank, weight, and index. Stability can be measured in two steps. The first step involves applying different amounts of perturbation (e.g., perturbation might be in the form of drawing training samples from the original data) into the training data at each iteration and obtaining the selected feature subsets. The second step involves using a stability measure to assess the similarities between the feature subsets. The majority of the stability measures compare selected feature subsets in pairs. Therefore, for n number of feature subsets, they perform $n(n-1)/2$ comparisons and the stability of the feature selection algorithm is simply the average of the stability results from each comparison. The greater the similarity between the resulting subsets, the greater the stability value.

7.1. Stability by Rank

The stability of the selection algorithms that rank features in terms of relevancy, e.g., Minimum Redundancy Maximum Relevance (mRMR) [47], is measured by evaluating the correlation (*or* similarity ratio) between ranking vectors using metrics, such as Spearman's Rank Correlation Coefficient (SRCC *or* Spearman's rho), Kendal's Rank Correlation Coefficient (KRCC *or* Kendall's tau), Canberra Distance, and Weighted Canberra Distance (see Table 3). In Table 3, X_i and Y_i represent i$^{\text{th}}$ ranked vectors, i.e., feature set, n is the total number of features, k is the top-k positions of the ranked feature set, CP represents the number of concordant pairs and DP represents the discordant ones.

Table 3. *The measures of stability by rank*

Measure	Formula	Bounds	Measures	Ref.
SRCC (*or* Spearman's ρ)	$\rho = 1 - \frac{6\sum_{i=1}^{n}(X_i - Y_i)^2}{(n^3 - n)}$	$-1 \leq \rho \leq 1$	Similarity	[46]
KRCC (*or* Kendall's τ)	$\tau = \frac{\#\,of\ CP\ -\#\,of\ DP}{\frac{n^2-1}{2}}$	$-1 \leq \tau \leq 1$	Similarity	[48]
Canberra Distance[8]	$d_{CD} = \sum_{i=1}^{n}\frac{\lvert X_i - Y_i\rvert}{\lvert X_i + Y_i\rvert}$	$[0, \infty]$	Dissimilarity	[49]
Weighted Canberra Distance[8]	$d_{WCD} = \sum_{i=1}^{n}\frac{\lvert min(X_i, k+1) - min(Y_i, k+1)\rvert}{min(X_i, k+1) + min(Y_i, k+1)}$	$[0, \infty]$		

7.2. Stability by Weight (*or* Weight Score)

The stability of the selection algorithms that weight features in terms of relevancy, e.g., ReliefF [50], is measured by evaluating the correlation (*or* similarity ratio) between weight vectors using Pearson's Correlation Coefficient (PCC). PCC (see Table 4) is the only stability measure in this category. Feature

[8] For Canberra and Weighted Canberra Distances $1 - d_{CD}$ or $1 - d_{WCD}$ gives similarity. Canberra and Weighted Canberra Distances can be bounded in the interval of [0,1] by dividing by the total number of features.

weights range from 0 to 1 or -1 to 1, where 1 denotes the most relevant attribute and 0 or -1 denotes the least relevant one. In Table 4, X_i and Y_i represent i$^{\text{th}}$ weight vectors, i.e., feature subsets, and $\bar{X}$ and $\bar{Y}$ represent sample means.

Table 4. *The measures of stability by weight*

Measure	Formula	Bounds	Measures	Ref.
Pearson's Correlation Coefficient (PCC)	$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})\,(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2 \sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$	$-1 \leq r \leq 1$	Similarity	[46]

7.3. Stability by Index (*or* Subset)

Feature selection algorithms, such as SBS (Sequential Backward Selection), represent their output as a vector of feature indices, e.g., $f = \{1,3,6,7,\ldots\}$, or as a binary vector, e.g., $f = [1,0,1,0,0,1,1,0,\ldots]$ (selected attributes are represented by 1). Therefore, measures in this category use set-based similarity. Some examples of stability by index measures are Sørensen-Dice Coefficient, Kuncheva and Jaccard Index, and Hamming Distance (see Table 5). In Table 5, X and Y represent index vectors, i.e., feature indexes, n is the total number of features, and c is the cardinality of feature subsets.

Table 5. *The measures of stability by index*

Measure	Formula	Bounds	Measures	Ref.
Jaccard Coefficient	$J(X,Y) = \frac{\lvert X \cap Y \rvert}{\lvert X \cup Y \rvert}$	[0,1]	Similarity	[51]
Jaccard Distance	$d_J = 1 - J(X,Y)$		Dissimilarity	
Sørensen-Dice Coefficient	$SD(X,Y) = \frac{2\lvert X \cap Y \rvert}{\lvert X \rvert + \lvert Y \rvert}$	[0,1]	Similarity	[52]
Jaccard Coefficient using Sørensen-Dice	$J(X,Y) = \frac{SD(X,Y)}{2 - SD(X,Y)}$	[0,1]	Similarity	
Sørensen-Dice using Jaccard Coefficient	$SD(X,Y) = \frac{2J(X,Y)}{1 + J(X,Y)}$	[0,1]		
Kuncheva Index	$KI(X,Y) = \frac{\lvert X \cap Y \rvert n - c^2}{nc - c^2}$	$[-1,1]$	Similarity	[53]
Hamming Distance	$d_H = \#(X \neq Y)$	—	Dissimilarity	[54]
Normalized Hamming Similarity	$H(X,Y) = 1 - \frac{d_H}{n}$	[0,1]	Similarity	
Lustgarten's Measure	$L(X,Y) = \frac{\lvert X \cap Y \rvert - \frac{\lvert X \rvert \lvert Y \rvert}{n}}{min\,(\lvert X \rvert, \lvert Y \rvert) - max\,(0, \lvert X \rvert + \lvert Y \rvert - n)}$	$[-1,1]$	Similarity	[55]
Ochiai Similarity	$O(X,Y) = \frac{\lvert X \cap Y \rvert}{\sqrt{\lvert X \rvert}\sqrt{\lvert Y \rvert}}$	[0,1]	Similarity	[56]
POG[9]	$POG(X,Y) = \frac{\lvert X \cap Y \rvert}{\lvert X \rvert}$	[0,1]	Similarity	[57]
nPOG[10]	$nPOG(X,Y) = \frac{\lvert X \cap Y \rvert - \frac{\lvert X \rvert \lvert Y \rvert}{n}}{\lvert X \rvert - \frac{\lvert X \rvert \lvert Y \rvert}{n}}$	$[1-n, 1]$	Similarity	[58]
Wald's Measure	$L(X,Y) = \frac{\lvert X \cap Y \rvert - \frac{\lvert X \rvert \lvert Y \rvert}{n}}{min\,(\lvert X \rvert, \lvert Y \rvert) - \frac{\lvert X \rvert \lvert Y \rvert}{n}}$	$[1-n, 1]$	Similarity	[59]

[9] POG is the acronym for the "Percentage of Overlapping Genes/Features".

[10] nPOG is the acronym for the "Normalized Percentage of Overlapping Genes/Features".

7.4. Other Types of Stability Measures

Gulgezen et al. [60] proposed a stability evaluation measure that uses a weighted bipartite graph and Symmetrical Uncertainty. The authors used Symmetric Uncertainty to assign weights to selected feature subsets. Unlike other stability measurement methods, the proposed approach assesses the similarity between feature values instead of feature indices. Symmetrical Uncertainty is an entropy-based nonlinear correlation that returns results in the range of 0 and 1 and can measure the association between dependent (features) and independent (class) variables. Therefore, it can also be used as a filter method. In Table 6, X and Y represent selected feature vectors, and IG and E are the information gain and the entropy, respectively.

Table 6. *The formula of Symmetrical Uncertainty*

Measure	Formula	Bounds	Measures	Ref.
Symmetrical Uncertainty	$SU(X,Y) = 2\left[\frac{IG(X\|Y)}{E(X)+E(Y)}\right]$ $where\ IG(X\|Y) = E(X) - E(X\|Y)\ and$ $E(X) = -\sum_i P(x_i)\, log_2(P(x_i))$ $E(X\|Y) = -\sum_j P(y_j) \sum_i P(x_i\|y_j)\, log_2(P(x_i\|y_j))$	[0,1]	Similarity	[60]

The stability measures mentioned above evaluate the amount of overlap between the result sets by using pairwise comparisons. Frequency-based stability metrics use the occurrence or occurrence frequency of an attribute or set of attributes. For the measures in this category, selected feature subsets should be represented as a binary matrix (where 1 means the feature is selected), where each row represents a subset and each column represents the selection of a particular feature. Some examples of frequency-based stability methods are Nogueira's measure [61], Lausser's measure [62], Entropy-based stability measure [63], Jensen-Shannon Divergence-based stability measure [64], corrected frequency of selection [65], and average frequency of selection [66]. Formulas and explanations of these metrics can be found in Nogueira's related work [61].

7.5. The Properties of Stability Measures

Kuncheva [53] identified the properties of a stability measure should have as *monotonicity, limits,* and *correction for chance*. In their study, Nogueira and Brown [67] summarized the desirable properties of a stability measure and presented which stability measure satisfies which property in a tabular form. These properties are:

1. **Monotonicity:** The stability result should increase as the similarity between the selected feature subsets increases.
2. **Having limits:** The result of the stability metric should be bounded between constants, e.g., the results of most stability metrics are in the range of $[0,1]$ or $[-1,1]$.
3. **Correction for chance:** In her study, Kuncheva [53] stated that *"the index should have a constant value for independently drawn subsets of features of the same cardinality*". As the cardinality of the selected feature subsets increases, the amount of overlap between the sets increases as well. This is called an intersection by chance. In such a case, the stability metric should have a constant that corrects the result. For example, Kuncheva Index satisfies this property.
4. **Symmetry:** Let f_1 and f_2 denote two different selected feature subsets, and s denote a stability measure. The stability value of $s(f_1, f_2)$ should be equal to $s(f_2, f_1)$.
5. **Independent of cardinality:** The stability metric should be used with subsets of selected features of different sizes.
6. **Maximum and minimum:** The stability metric should reach its maximum if the selected feature subsets are the same, and its minimum if the selected feature subsets are completely different.

8. REASONS AND SOLUTIONS FOR INSTABILITY

Data characteristics (quality) and bias-variance decomposition influence the stability of feature selection algorithms. Bias-variance decomposition is a method used to analyze expected generalization errors in supervised learning. In supervised learning, i.e., classification and regression, bias and variance are two sources of error that cause the performance deterioration of machine learning algorithms. Sometimes learners are failing to capture the relevant associations between the predictors, i.e., features, and the outputs, i.e., class variable *or* output, due to erroneous assumptions. This is known as a bias error. Variance error is the measure of the variability of the learner against training data perturbations. A better tradeoff between the bias and the variance can increase the stability of feature selection algorithms [68]. Besides bias and variance, another factor that badly influences the performance of the learners is an irreducible error. Irreducible error is the random noise (error) in the problem itself (data-dependent) and cannot be reduced at all.

Several studies have argued that data-dependent issues such as distribution, imbalanced datasets, sparse data, sample size, and the number of features (high dimension-low sample size) play an important role in stability [61, 69-72]. Therefore, the characteristics of data can be summarized using descriptive statistics, which include central tendency and variability measures, to detect and avert any negative effects. Central tendency is measured using mean, median, and mode, whereas standard deviation, variance, the value range (minimum and maximum) of the variables, kurtosis, and skewness are used to measure variability, i.e., dispersion. Another reason for instability is the algorithms themselves. Stability can be maintained by data variance reduction [73], ensemble (see Section 4.5) and group-based feature selection approaches [74-76]. Group-based feature selection approaches use highly correlated features and selects features in a grouped manner to improve both the stability and prediction performance of models.

9. CONCLUSION

The present research reviews the literature on feature selection and feature selection stability. Problems caused by high-dimensional datasets have raised interest in dimension, i.e., data, reduction approaches, like feature selection. For this reason, over the years, a large pool of feature selection techniques has emerged. Since these techniques use different strategies to select relevant features, choosing the appropriate method has a pivotal role in feature selection process. Various studies have proved that eliminating irrelevant and redundant features helps to improve the performance of machine learning algorithms and the quality of the data analysis. However, feature selection adds extra complexity to the learning process, and it is not always feasible to select an optimal feature set, especially when they are closely related.

The quality of selection algorithms is determined by the models built using the feature subsets they select, as well as their stability. Stability refers to the robustness, i.e., insensitivity, of the selection algorithm to the minor changes in the training set. Stable feature selection methods produce repetitive results. The stability of the selection algorithm is an important issue because unstable algorithms mislead the user in selecting the resulting subset of attributes and undermine confidence in the algorithm and analysis process. In this study, we also addressed different sources of instability and metrics used to assess the stability of the feature selection algorithms.

Supervised feature selection and supervised feature selection stability are major areas of interest within the field of feature selection. A limited number of studies in the literature have analyzed unsupervised and semi-supervised feature selection stability. Thus, these topics constitute an open research field for researchers. This study is mostly based on supervised feature selection and selection stability. Unfortunately, we were unable to encompass other types of feature selection in detail due to the wide scope of the subject. For this reason, we refer the readers to the cited studies to obtain an understanding of the topics that were not covered in detail.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

[1] Kohavi, R., John, G.H., “Wrappers for feature subset selection”, Artificial Intelligence, 97(1-2): 273-324, (1997).

[2] Yu, L., Liu, H., “Efficient Feature Selection via Analysis of Relevance and Redundancy”, Journal of Machine Learning Research, 5: 1205-1224, (2004).

[3] Yu, L., Liu, H., “Redundancy Based Feature Selection for Microarray Data”, KDD ‘04: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 737-742, (2004).

[4] Cho, S.-B., Won, H.-H., “Machine Learning in DNA Microarray Analysis for Cancer Classification”, APBC ‘03: Proceedings of the First Asia-Pacific Bioinformatics Conference on Bioinformatics, Adelaide, SA, Australia, 19: 189-198, (2003).

[5] Tang, J., Zhou, S., “A new approach for feature selection from microarray data based on mutual information”, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 13(6): 1004-1015, (2016).

[6] Inza, I., Larranaga, P., Blanco, R., Cerrolaza, A.J., “Filter versus wrapper gene selection approaches in DNA microarray domains”, Artificial Intelligence in Medicine, 31(2): 91-103, (2004).

[7] Yang, Q., Jia, X., Li, X., Feng, J., Li, W., Lee, J., “Evaluating feature selection and anomaly detection methods of hard drive failure prediction”, IEEE Transactions on Reliability, 70(2): 749-760, (2021).

[8] Lee, W., Stolfo, S.J., Mok, K.W., “Adaptive intrusion detection: a data mining approach”, Artificial Intelligence Review, 14: 533-567, (2000).

[9] Alazab, A., Hobbs, M., Abawajy, J., Alazab, M., “Using Feature Selection for Intrusion Detection System”, International Symposium on Communications and Information Technologies (ISCIT), Gold Coast, QLD, Australia, 296-301, (2012).

[10] Huang, K., Aviyente, S., “Wavelet feature selection for image classification”, IEEE Transactions on Image Processing, 17(9): 1709-1720, (2008).

[11] Dy, J.G., Brodley, C.E., Kak, A., Broderick, L.S., Aisen, A.M., “Unsupervised feature selection applied to content-based retrieval of lung images”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(3): 373-378, (2003).

[12] Forman, G., “An Extensive Empirical Study of Feature Selection Metrics for Text Classification”, Journal of Machine Learning Research, 3: 1289-1305, (2003).

[13] Jing, L.-P., Huang, H.-K., Shi, H.-B., “Improved Feature Selection Approach TFIDF in Text Mining”, Proceedings of the International Conference on Machine Learning and Cybernetics, Beijing, China, 944-946, (2002).

[14] Bai, X., Gao, X., Xue, B., “Particle swarm optimization based two-stage feature selection in text mining”, 2018 IEEE Congress on Evolutionary Computation (CEC), 1-8, (2018).

[15] Fisher, R.A., “The use of multiple measurements in taxonomic problems”, Annals of Eugenics, 7: 179-188, (1936).

[16] Han, D., Kim, J., “Unified simultaneous clustering and feature selection for unlabeled and labeled data”, IEEE Transactions on Neural Networks and Learning Systems, 29(12): 6083-6098, (2018).

[17] Zhao, Z., Liu, H., “Spectral Feature Selection for Supervised and Unsupervised Learning”, ICML ‘07: Proceedings of the 24th International Conference on Machine Learning, Corvalis, OR, USA, 1151-1157, (2007).

[18] Tang, J., Alelyani, S., Liu, H., “Feature selection for classification: a review”, Data Classification: Algorithms and Applications, CRC Press, 37-64, (2014).

[19] Ang, J.C., Mirzal, A., Haron, H., Hamed, H.N.A., “Supervised, unsupervised, and semi-supervised feature selection: a review on gene selection”, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 13(5): 971-989, (2015).

[20] Yang, W., Wang, K., Zuo, W., “Neighborhood Component Feature Selection for High-Dimensional Data”, Journal of Computers, 7(1): 161-168, (2012).

[21] Dy, J.G., Brodley, C.E., Wrobel, S. (Editor), “Feature Selection for Unsupervised Learning”, The Journal of Machine Learning Research, 5: 845-889, (2004).

[22] Solorio-Fernandez, S., Carrasco-Ochoa, J.A., Martinez-Trinidad, J.F., “A review of unsupervised feature selection methods”, Artificial Intelligence Review, 53: 907-948, (2020).

[23] Boutsidis, C., Mahoney, M.W., Drineas, P., “Unsupervised Feature Selection for Principal Components Analysis”, Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, 61-69, (2008).

[24] He, X., Cai, D., Niyogi, P., “Laplacian Score for Feature Selection”, NIPS ‘05: Proceedings of the 18th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 507-514, (2005).

[25] Zhao, Z., Liu, H., “Semi-supervised Feature Selection via Spectral Analysis”, Proceedings of the 7th SIAM International Conference on Data Mining, Minneapolis, MN, USA, 641-646, (2007).

[26] Ren, J., Qiu, Z., Fan, W., Cheng, H., Yu, P.S., “Forward semi-supervised feature selection”, PAKDD ‘08: Advances in Knowledge Discovery and Data Mining, Lecture Notes in Computer Science, 5012: 970-976, (2008).

[27] Sheikhpour, R., Sarram, M.A., Gharaghani, S., Chahooki, M.A.Z., “A Survey on semi-supervised feature selection methods”, Pattern Recognition, 64: 141-158, (2017).

[28] Xu, Z., King, I., Lyu, M.R., Jin, R., “Discriminative semi-supervised feature selection via manifold regularization”, IEEE Transactions on Neural Networks, 21(7): 1303-1308, (2010).

[29] Zhao, J., Lu, K., He, X., “Locality sensitive semi-supervised feature selection”, Neurocomputing, 71(10-12): 1842-1849, (2008).

[30] Guyon, I., Elisseeff, A., Kaelbling, L.P. (Editor), “An Introduction to Variable and Feature Selection”, Journal of Machine Learning Research, 3: 1157-1182, (2003).

[31] Haury, A.-C., Gestraud, P., Vert, J.-P., “The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures”, PLoS ONE, 6(12): e28210, (2011).

[32] Breiman, L., Friedman, J.H., Stone, C.J., Olshen, R.A., “Classification and regression trees”, 1st Ed., United Kingdom: Chapman and Hall/CRC, 18-55, 216-264, (1984).

[33] Quinlan, J.R., “Induction of decision trees”, Machine Learning, 1: 81-106, (1986).

[34] Tharwat, A., “Classification assessment methods: a detailed tutorial”, Applied Computing and Informatics, (2018).

[35] Landgrebe, T.C.W., Duin, R.P.W., “Approximating the multiclass ROC by pairwise analysis”, Pattern Recognition Letters, 28(13): 1747-1758, (2007).

[36] Fawcett, T., “An introduction to ROC analysis”, Pattern Recognition Letters, 27(8): 861-874, (2006).

[37] Turney, P., “Technical note: bias and the quantification of stability”, Machine Learning, 20, 23-33, (1995).

[38] Hulse, J.V., Khoshgoftaar, T.M., Napolitano, A., Wald, R., “Feature Selection with High-Dimensional Imbalanced Data”, 2009 IEEE International Conference on Data Mining Workshops, Miami, FL, USA, 507-514, (2009).

[39] Maldonado, S., Weber, R., Famili, F., “Feature selection for high-dimensional class-imbalanced data sets using support vector machines”, Information Sciences, 286: 228-246, (2014).

[40] Viegas, F., Rocha, L., Gonçalves, M., Mourao, F., Sa, G., Salles, T., Andrade, G., Sandin, I., “A genetic programming approach for feature selection in highly dimensional skewed data”, Neurocomputing, 273: 554-569, (2018).

[41] Katrutsa, A., Strijov, V., “Comprehensive study of feature selection methods to solve multicollinearity problem according to evaluation criteria”, Expert Systems with Applications, 76: 1-15, (2017).

[42] Jain, A., Zongker, D., “Feature selection: evaluation, application, and small sample performance”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(2): 153-158, (1997).

[43] Wu, X., Cheng, Q., “Algorithmic Stability and Generalization of an Unsupervised FSA”, NeurIPS 2021: 35th Conference on Neural Information Processing Systems, 1-14, (2021).

[44] Helleputte, T., Dupont, P., “Partially Supervised Feature Selection with Regularized Linear Models”, ICML ‘09: Proceedings of the 26th Annual International Conference on Machine Learning, 409-416, (2009).

[45] Lai, D.T.C., Garibaldi, J.M., “Improving Semi-supervised Fuzzy C-Means Classification of Breast Cancer Data Using Feature Selection”, 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Hyderabad, India, 1-8, (2013).

[46] Kalousis, A., Prados, J., Hilario, M., “Stability of feature selection algorithms: a study on high-dimensional spaces”, Knowledge and Information Systems, 12: 95-116, (2007).

[47] Ding, C., Peng, H., “Minimum Redundancy Feature Selection from Microarray Gene Expression Data”, Journal of Bioinformatics and Computational Biology, 3(2): 185-205, (2005).

[48] Shabbir, A., Javed, K., Ansari, Y., Babri, H.A., “Stability of Feature Ranking Algorithms on Binary Data”, Pakistan Journal of Engineering and Applied Sciences, 15: 76-86, (2014).

[49] Jurman, G., Merler, S., Barla, A., Paoli, S., Galea, A., Furlanello, C., "Algebraic stability indicators for ranked lists in molecular profiling", Bioinformatics, 24(2): 258-264, (2008).

[50] Kononenko, I., Simec, E., Robnik-Sikonja, M., "Overcoming the myopia of inductive learning algorithms with RELIEFF", Applied Intelligence, 7: 39-55, (1997).

[51] Saeys, Y., Abeel T., Van de Peer, Y., "Robust feature selection using ensemble feature selection techniques", ECML PKDD '08: Machine Learning and Knowledge Discovery in Databases, 5212: 313-325, (2008).

[52] Yu, L., Ding, C., Loscalzo, S., "Stable Feature Selection via Dense Feature Groups", KDD '08: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, 803-811, (2008).

[53] Kuncheva, L.I., "A Stability Index for Feature Selection", Proceedings of the 25th IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria, 390-395, (2007).

[54] Dunne, K., Cunningham, P., Azuaje, F., "Solutions to Instability Problems with Sequential Wrapper-based Approaches to Feature Selection", Journal of Machine Learning Research, 1-22, (2002).

[55] Lustgarten, J.L., Gopalakrishnan, V., Visweswaran, S., "Measuring Stability of Feature Selection in Biomedical Datasets", AMIA '09: Annual Symposium Proceedings, Published Online, 406-410, (2009).

[56] Zucknick, M., Richardson, S., Stronach, E.A., "Comparing the characteristics of gene expression profiles derived by univariate and multivariate classification methods", Statistical Applications in Genetics and Molecular Biology, 7(1): 1-28, (2008).

[57] Shi, L., Tong, W., Fang, H., Scherf, U., Han, J., Puri, R.K., Frueh, F.W., Goodsaid, F.M., Guo, L., Su, Z., Han, T., Fuscoe, J.C., Xu, Z.A., Patterson, T.A., Hong, H., Xie, Q., Perkins, R.G., Chen, J.J., Casciano, D.A., "Cross-platform comparability of microarray technology: intraplatform consistency and appropriate data analysis procedures are essential", BMC Bioinformatics 6, Article number S12, (2005).

[58] Zhang, M., Zhang, L., Zou, J., Yao, C., Xiao, H., Liu, Q., Wang, J., Wang, D., Wang, C., Guo, Z., "Evaluating reproducibility of differential expression discoveries in microarray studies by considering correlated molecular changes", Bioinformatics, 25(13): 1662-1668, (2009).

[59] Wald, R., Khoshgoftaar, T., Dittman, D., "A New Fixed-overlap Partitioning Algorithm for Determining Stability of Bioinformatics Gene Rankers", 11th International Conference on Machine Learning and Applications (ICMLA), Boca Raton, FL, USA, 170-177, (2012).

[60] Gulgezen, G., Cataltepe, Z., Yu., L., "Stable and accurate feature selection", ECML PKDD '09: Machine Learning and Knowledge Discovery in Databases, 5781: 455-468, (2009).

[61] Nogueira, S., "Quantifying the stability of feature selection", Ph.D. Thesis, University of Manchester, Manchester, United Kingdom, 21-67, (2018).

[62] Lausser, L., Müssel, C., Maucher, M., Kestler, H.A., "Measuring and visualizing the stability of biomarker selection techniques", Computational Statistics, 28: 51-65, (2013).

[63] Krizek, P., Kittler, J., Hlavac, V., "Improving Stability of Feature Selection Methods", 12th International Conference on Computer Analysis of Images and Patterns (CAIP), Vienna, Austria, 929-936, (2007).

[64] Guzman-Martinez, R., Alaiz-Rodriguez, R., "Feature selection stability assessment based on the Jensen-Shannon divergence", Lecture Notes in Computer Science, 6911: 597-612, (2011).

[65] Davis, C.A., Gerick, F., Hintermair, V., Friedel, C.C., Fundel, K., Küffner, R., Zimmer, R., "Reliable gene signatures for microarray classification: assessment of stability and performance", Bioinformatics, 22(19): 2356-2363, (2006).

[66] Goh, W.W.B., Wong, L., "Evaluating Feature Selection Stability in Next-Generation Proteomics", Journal of Bioinformatics and Computational Biology, 14(5): 1650029, (2016).

[67] Nogueira, S., Brown, G., "Measuring the stability of feature selection", ECML PKDD '16: Machine Learning and Knowledge Discovery in Databases, 9852: 442-457, (2016).

[68] Munson, M.A., Caruana, R., "On feature selection, bias-variance, and bagging", ECML PKDD '09: Machine Learning and Knowledge Discovery in Databases, 5782: 144-159, (2009).

[69] Alelyani, S., "On feature selection stability: a data perspective", Ph.D. Thesis, Arizona State University, Phoenix, USA, 10-57, (2013).

[70] Alelyani, S., Liu, H., Wang, L., "The Effect of the Characteristics of the Dataset on the Selection Stability", 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, USA, 970-977, (2011).

[71] Dittman, D., Khoshgoftaar, T., Wald, R., Napolitano, A., "Similarity Analysis of Feature Ranking Techniques on Imbalanced DNA Microarray Datasets", 2012 IEEE International Conference on Bioinformatics and Biomedicine, Philadelphia, PA, USA, 1-5, (2012).

[72] Alelyani, S., Zhao, Z., Liu, H., "A Dilemma in Assessing Stability of Feature Selection Algorithms", 2011 IEEE International Conference on High Performance Computing and Communications, Banff, AB, Canada, 701-707, (2011).

[73] Han, Y., Yu, L., "A Variance Reduction Framework for Stable Feature Selection", 2010 IEEE International Conference on Data Mining, Sydney, NSW, Australia, 206-215, (2010).

[74] Kamkar, I., "Building stable predictive models for healthcare applications: a data-driven approach", Ph.D. Thesis, Deakin University, Geelong, Australia, 34-52, (2016).

[75] Tang, F., Adam, L., Si, B., "Group feature selection with multiclass support vector machine", Neurocomputing, 317: 42-49, (2018).

[76] Loscalzo, S., Yu, L., Ding, C.H.Q., "Consensus Group Stable Feature Selection", Conference: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 567-575, (2009).