applied sciences

MDPI

Article

Framework for the Ensemble of Feature Selection Methods

Maritza Mera-Gaona [1,*], Diego M. López [1], Rubiel Vargas-Canas [1] and Ursula Neumann [2]

[1] Faculty of Electronic Engineering and Telecommunications, Campus Tulcan, University of Cauca, Popayán 190001, Colombia; dmlopez@unicauca.edu.co (D.M.L.); rubiel@unicauca.edu.co (R.V.-C.)

[2] Group Data Science, Division Supply Chain Services SCS, Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS, 90411 Nuremberg, Germany; ursulaneuman@iis.fraunhofer.de

* Correspondence: maritzag@unicauca.edu.co

Abstract: Feature selection (FS) has attracted the attention of many researchers in the last few years due to the increasing sizes of datasets, which contain hundreds or thousands of columns (features). Typically, not all columns represent relevant values. Consequently, the noise or irrelevant columns could confuse the algorithms, leading to a weak performance of machine learning models. Different FS algorithms have been proposed to analyze highly dimensional datasets and determine their subsets of relevant features to overcome this problem. However, very often, FS algorithms are biased by the data. Thus, methods for ensemble feature selection (EFS) algorithms have become an alternative to integrate the advantages of single FS algorithms and compensate for their disadvantages. The objective of this research is to propose a conceptual and implementation framework to understand the main concepts and relationships in the process of aggregating FS algorithms and to demonstrate how to address FS on datasets with high dimensionality. The proposed conceptual framework is validated by deriving an implementation framework, which incorporates a set of Phyton packages with functionalities to support the assembly of feature selection algorithms. The performance of the implementation framework was demonstrated in several experiments discovering relevant features in the Sonar, SPECTF, and WDBC datasets. The experiments contrasted the accuracy of two machine learning classifiers (decision tree and logistic regression), trained with subsets of features generated either by single FS algorithms or the set of features selected by the ensemble feature selection framework. We observed that for the three datasets used (Sonar, SPECTF, and WD), the highest precision percentages (86.95%, 74.73%, and 93.85%, respectively) were obtained when the classifiers were trained with the subset of features generated by our framework. Additionally, the stability of the feature sets generated using our ensemble method was evaluated. The results showed that the method achieved perfect stability for the three datasets used in the evaluation.

Keywords: feature selection; variable elimination; relevant features; consensus; ensemble learning; ensemble feature selection

Citation: Mera-Gaona, M.; López, D.M.; Vargas-Canas, R.; Neumann, U. Framework for the Ensemble of Feature Selection Methods. *Appl. Sci.* **2021**, *11*, 8122. https://doi.org/10.3390/app11178122

Academic Editor: Andrea Prati

Received: 30 July 2021
Accepted: 19 August 2021
Published: 1 September 2021

1. Introduction

A feature is defined as a measurable property of a process or entity that is being observed. It is also known as attribute, component, variable, column, or dimension [1]. In the field of machine learning, a set of features describes a domain and classifies, detects, or recognizes patterns. In the past, few machine learning applications used more than 40 features [2]; however, nowadays, the number of features has increased their size from tens to hundreds of features. Consequently, handling this information is costly due to its processing, thus requiring more time and resources. In this sense, many studies in the last two decades faced this problem, especially when datasets have a high number of features and few instances. This problem is called "the curse of dimensionality" [3].

Consequently, in 1997 [4,5], the authors described the first studies about feature selection (FS) in domains where it is common to find datasets with several dozens of features. Since then, many techniques have been developed to solve the problems generated by the number of features. According to the findings, most of the features in those datasets are redundant or irrelevant [6]. Given this, FS techniques focus on identifying features with high differentiating power while discarding those considered irrelevant or redundant. Thereby, the main goal of FS is to avoid features that do not efficiently allow generalization in processes of classification, detection, or pattern recognition

1.1. Dataset's Growth

Recently, datasets with large numbers of features are more frequent in different domains. Three of the most representative examples are microarray classification, text categorization, and signal classification. In the first case, developments in DNA microarray have generated numerous datasets with this kind of data. In most of these datasets, the number of instances is not higher than 100 (patients), and the number of features (genes) ranges from 6.000 to 60.000 [7]. However, previous studies showed that most of the genes in these datasets do not represent helpful information to support a machine learning process. Consequently, a preprocessing stage is needed to efficiently classify microarray data [8,9]. In [10,11], the authors describe how to reduce computational cost and improve performance in the classification of microarrays by selecting a representative subset of genes from the original set.

Likewise, in-text categorization documents are represented by an array built from their vocabulary and the frequencies of words in such documents. Those vocabulary sets have hundreds of thousands of words. However, vocabulary is pruned to remove the least essential words from the documents in an initial stage. Thus, the size of the array that represents the documents is reduced. In literature, there are several collections of documents used in different application domains, for instance, email analysis [12], detection of articles related to terrorism on the web [13], automatic classification of text [14], opinions [13], feelings [15,16], and emotions [17], among others. These collections have between 5000 and 800,000 documents.

In the field of signal classification, previous works have used many mechanisms to process the signal and get a set of features capable of describing the signal. These features are used to classify or detect patterns. In the medical domain, the high availability of devices designed to capture biosignals has supported the diagnosis of diseases by identifying normal and abnormal patterns in the signals. Hence, several authors have developed solutions to support the automatic analysis of signals such as EEG and ECG. In the automated analysis of EEGs, for instance, the signals considered are multichannel with (i) information in channels, which range from 12 to 64 in number, (ii) duration between 20 min and 72 h, and (iii) a sampling rate between 100 and 256 per second. Considering the above, analyzing an EEG signal is a complex task since it contains much information. Thus, each channel from the signal is divided into segments, and many feature extractors must be applied to describe them. The process to segment each channel of the signal allows the identification of abnormalities, which appear in short periods [18].

In [19], the authors analyzed the current context of "Big Data" and "Big Dimensionality." They introduced those concepts to explain how to handle datasets with unbalanced data, noise, few instances, and a high number of features. They found out that the dataset sizes are not growing in both dimensions, columns, and rows. Besides, the most important repositories of datasets used in the experiments of machine learning contain datasets with thousands or millions of features. In many cases, the number of features widely surpasses the number of rows. For example, in the UCI machine learning repository [20], there are 18 datasets with more than 5000 features. The LIBSVM database contains datasets with over a million features [21]. Therefore, researchers have focused on developing methods to reduce the size of datasets using a set of objective criteria. This allows them to represent the complex original dataset in a simple dataset.

1.2. Context of Ensemble Feature Selection

Depending on the design of FS techniques, they are classified into three types of methods: filters, wrappers, and embedded. Each type defines advantages or disadvantages that are directly related to the context of the dataset. In general, these three types of FS techniques face typical problems, namely, (i) they have a good performance on a dataset, but by adding or removing instances, the performance decreases, (ii) they allow the removal of features quickly, but they are not capable of detecting redundant features, (iii) they need to have a correctly balanced dataset, and (iv) their performance is affected by the presence of noise in the data.

Moreover, there is a large number of FS methods. However, there are no tools or solutions to determine objectively the algorithms, which would work best with the data of a particular domain. Therefore, in some studies, authors have used a trial-and-error scheme. They have tested different FS algorithms using one or more classifiers and then chosen the one with the best performance in the test.

Alternatively, ensemble learning approaches have been proposed to select features based on the consensus or aggregation of several FS algorithms. For example, in [22], researchers proposed a classification algorithm based on K-Nearest Neighbors (KNN). They obtained and combined several outcomes from the KKN algorithm, and each result was obtained by using a different set of features. In 1998, the authors proposed an ensemble feature selection (EFS) method designed for decision trees [23]. In 1999 [24], an EFS method based on a genetic algorithm was proposed to improve the quality of the features used in the learners.

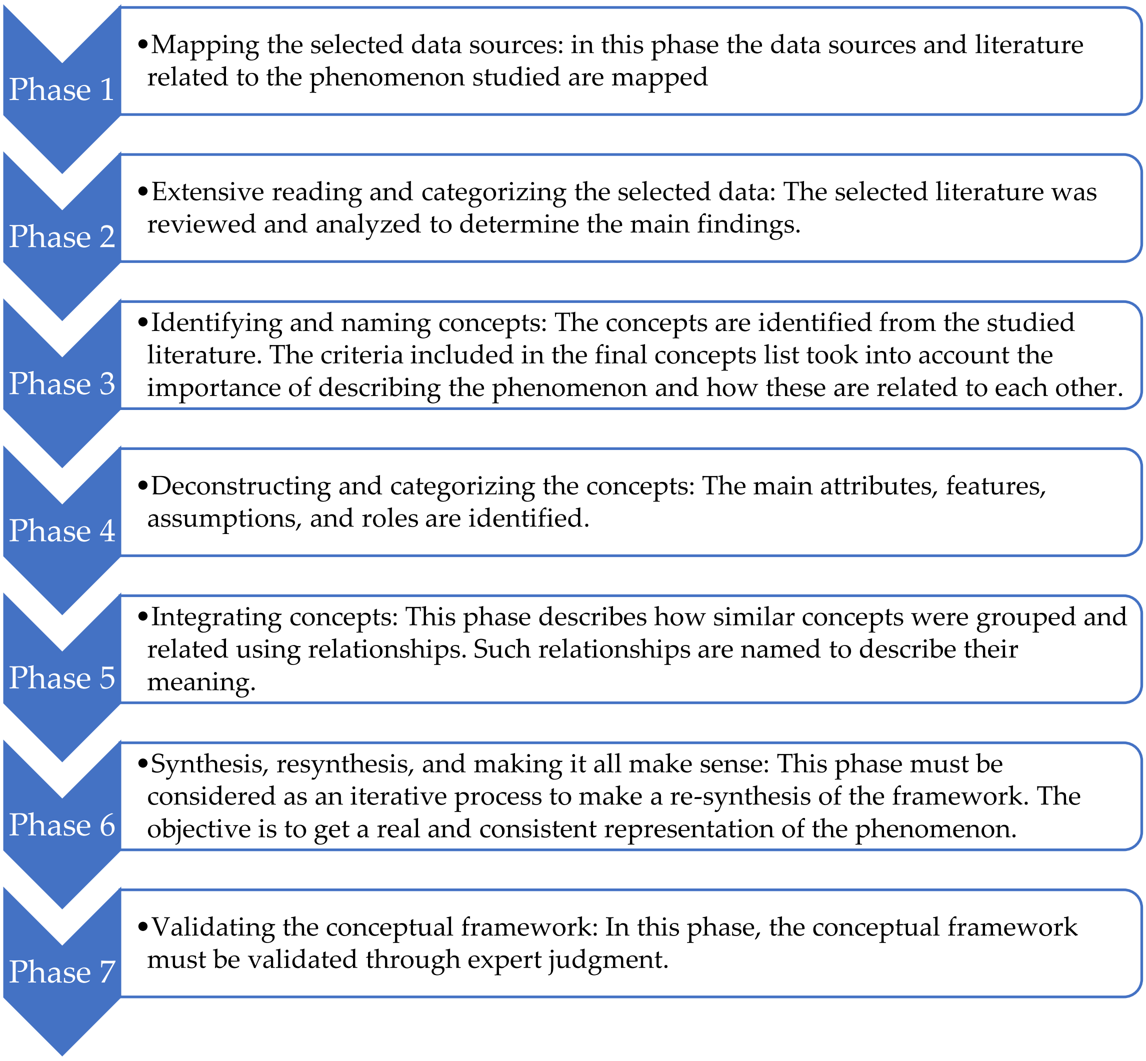
Recently, many studies have been conducted about EFS; some involve using classifiers while others do not. In [7], an EFS algorithm aggregates a set of FS algorithms based on filters to classify microarrays. The scheme in this study aims to use several filters and generate a subset of features for each filter. The subsets generated are used to train a classifier, and, subsequently, the outputs of classifiers are combined using simple voting. In [25], a mechanism of EFS on microarrays was built to determine relevant genes in the classification of cancer. A robust feature selection process is conducted in [26], the selection of features is based on EFS, and the findings showed that the approach obtained great promise for datasets with many features and few samples. The bi-objective genetic algorithm was used in [27] to develop a method of EFS. The evaluation showed that the proposal reached to obtain robustly and noise resilient subsets of features. An approach based on selecting in Random Forest and co-forest was implemented in [28]. The method allows selecting features in datasets with unlabeled data.

The main objective of this paper is to propose a framework to determine how to improve the FS on datasets with high dimensionality and a few instances. Furthermore, the framework seeks to guide the design of an EFS mechanism to gather the advantages of different FS algorithms, avoid their biases, and compensate for their disadvantages. For this, we designed a conceptual framework to understand the main concepts and relationships in the aggregation of a set of FS algorithms. Following this, an implementation framework was built to validate the theoretical proposal.

The rest of the document is organized as follows: Section 2 describes the qualitative method used to develop the proposed conceptual framework. Section 3 presents the implementation and evaluation of the framework. Section 4 offers a discussion of results and contributions. Finally, Section 5 describes the main conclusions of this research.

2. Materials and Methods

We followed the qualitative method described in [29] to propose a conceptual framework for our EFS. The method establishes a set of phases to design the framework as developing a plan or network of concepts linked together to describe a particular phenomenon. It represents a process to select a set of data sources, classify the data found, identify the main concepts related to them, and review and validate the proposal. The main objective is to highlight the sense and importance of the relationships that associate the concepts. Due to this, the concepts are considered a collection and a set of entities with a defined role. The methodology describes eight phases. However, in this study, only the first seven phases were used since the last one corresponds to the reformulation of the framework, which is included in phase seven for our research. The implementation built in this study validated the conceptual proposal. Thus, the improvements and adjustments implemented as part of the development process in phase seven represented the rethinking of the conceptual framework. Figure 1 shows the phases mentioned.



Phase 1
- Mapping the selected data sources: in this phase the data sources and literature related to the phenomenon studied are mapped

Phase 2
- Extensive reading and categorizing the selected data: The selected literature was reviewed and analyzed to determine the main findings.

Phase 3
- Identifying and naming concepts: The concepts are identified from the studied literature. The criteria included in the final concepts list took into account the importance of describing the phenomenon and how these are related to each other.

Phase 4
- Deconstructing and categorizing the concepts: The main attributes, features, assumptions, and roles are identified.

Phase 5
- Integrating concepts: This phase describes how similar concepts were grouped and related using relationships. Such relationships are named to describe their meaning.

Phase 6
- Synthesis, resynthesis, and making it all make sense: This phase must be considered as an iterative process to make a re-synthesis of the framework. The objective is to get a real and consistent representation of the phenomenon.

Phase 7
- Validating the conceptual framework: In this phase, the conceptual framework must be validated through expert judgment.

Figure 1. Methodology to Design Conceptual Framework.

2.1. Phase 1: Mapping the Selected Data Sources

In this case, the mapping considered the theory and research studies about Feature Selection and/or Ensemble Learning. Taking into account the above, the data source selected was the literature related to the following topics:

- Machine Learning
- Relevance Analysis
- Feature Selection
- Dimensionality Reduction
- Ensemble Learning
- Ensemble Feature Selection
- Consensus and aggregation

2.2. Phase 2: Extensive Reading and Categorizing the Selected Data

According to the analysis carried out, the research studies were classified into the following categories:

- Feature Selection
- Performance
- Classification
- Ensemble Learning
- Consensus-Aggregation

2.3. Phase 4: Deconstructing and Categorizing the Concepts

The main attributes, features, assumptions, and roles are represented in Section 3.

2.4. Phase 6: Synthesis, Re-Synthesis, and Making It All Make Sense

The graphic representation is shown in Section 3. This contains concepts, groupings, roles, and relationships.

2.5. Phase 7: Validating the Conceptual Framework

This phase recommends that the conceptual framework must be validated through expert judgment. However, to obtain a quantitative evaluation, the main concepts of the conceptual framework are implemented in Phyton. The evaluation shown in Section 3 describes the results with which F-EFS is validated.

3. Results

This section presents the design and implementation of a conceptual framework to support our EFS. This has been divided into three parts. The first part describes the theory about the main concepts considered to represent the framework. The second one describes the definition of the framework by associating the selected concepts to each other through relationships. Finally, the third one describes the implementation and evaluation of the framework.

3.1. Main Concepts

The different types of FS algorithms are described below: filters, wrappers, embedded, and the methods to aggregate experts' opinions.

- Filters: techniques that are easy to implement and can be scaled to use datasets with high dimensionality. Nonetheless, this type ignores the interaction with a classifier. For example, an $R(f)$ function evaluates the relevance of each feature, and the output of the filter algorithm corresponds to a ranking that orders the features according to $R(f)$ [30].
- Wrappers: methods, which evaluate the relevance of the subsets of features by using a classifier. Thereby, the best subset of features is selected by the learning algorithm. However, the computational cost of these techniques is high because when choosing the best subset, many subsets must be evaluated [5].
- Embedded: type of mechanism, which combines the advantages of the filters and wrappers. The main objective is to get the best performance in the learning process from a learning algorithm using a subset of features [2].
- Consensus: In ensemble learning, it is also called consensus theory of aggregation. Widely used in social sciences and administration, its main objective is to find a way to combine expert opinions through consensus rules [31].

3.2. Conceptual Framework

According to the qualitative method described in [29], a literature revision was conducted to select the data sources and understand the main concepts related to feature selection, ensemble learning, and consensus. Figure 2 shows the concepts and relationships identified from the data sources selected.

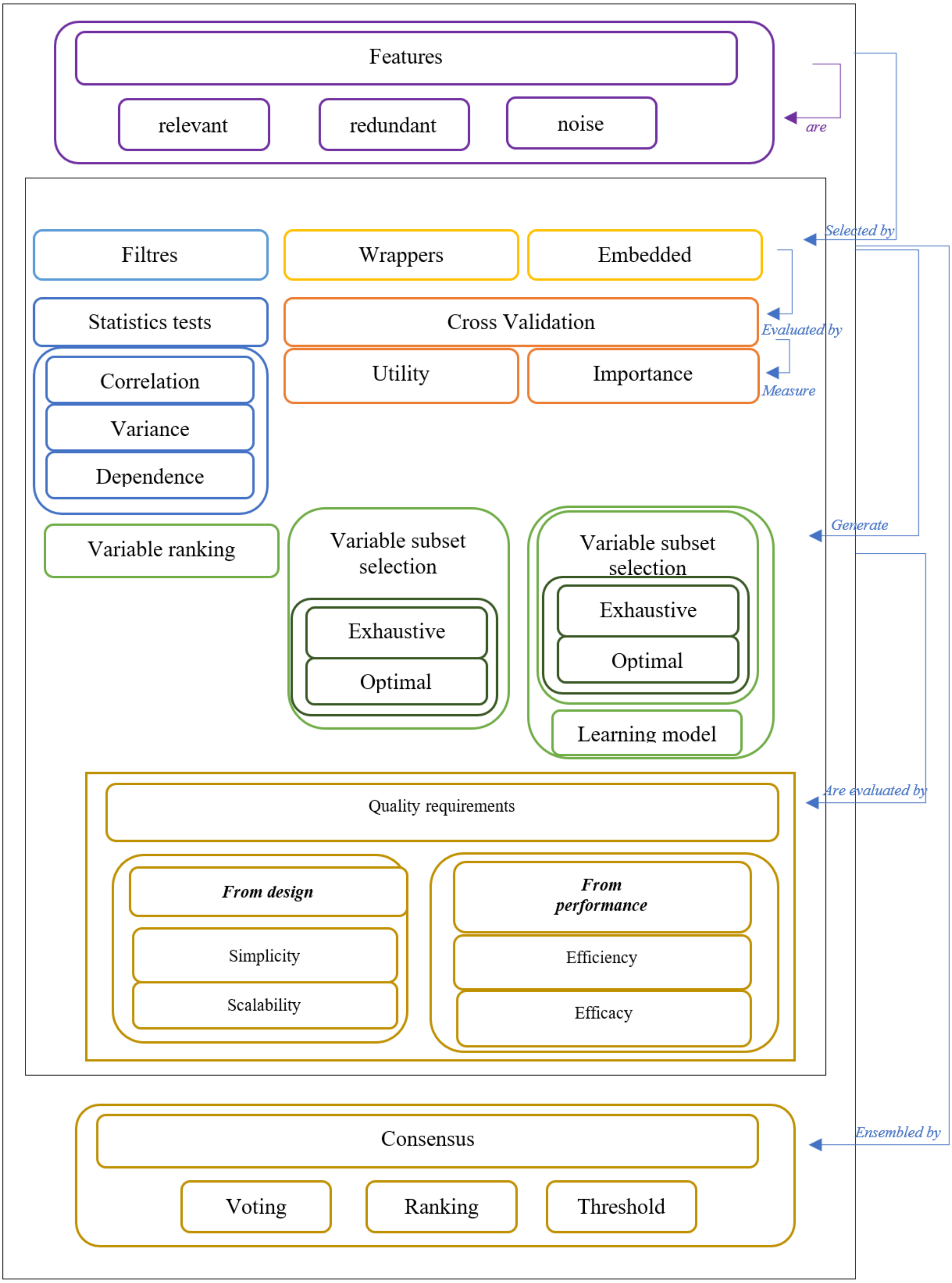


Figure 2. Conceptual framework.

The literature about relevance analysis, feature selection, and dimensionality reduction allowed the identification of the three types of methods to determine the features with higher differentiation power. Thus, the types of algorithms were analyzed considering their design, objective, and performance, which permitted the identification of the main concepts and relationships that describe the framework.

Considering the FS theory, the datasets contain three types of features: relevant, redundant, and noise. Also, some authors state that the relevant features have either solid or weak relevance [32]. All types of features can be identified by filters, wrappers, or embedded methods. According to the method, the focus can be on identifying features with low differentiation power, the dependence of features, and relevant features. For instance, to identify segments with low relevance, the algorithms analyze columns with low variance.

In general, the methods of FS use measures to evaluate the relevance of the features through statistical tests or cross-validation. The results obtained in these evaluations define the following: a ranking of feature relevance in the case of the filter-based methods, a subset of relevant features in the case of wrappers, or a subset of features with a learning model in the case of the embedded methods. In the wrappers or embedded methods, the FS is based on searching a subset of features by evaluating $\boldsymbol{n}$ subsets and selecting the one that achieves the best performance in the classification.

The outputs of the FS methods are evaluated from several perspectives. Considering the problem to solve, some criteria can be more or less relevant. Because of this, the framework establishes the evaluation of algorithms by reviewing either their performance or their design. In terms of performance, efficiency and effectiveness are evaluated by testing the subset of features selected in a classification process. In terms of design, simplicity and scalability are evaluated by the designers of the algorithms.

Following ensemble learning, the consensus of several experts improves the creation of a decision in a context [31]. Thereby, the conceptual framework considers the pooling of several FS algorithms through the consensus of a set of subsets of features selected by each method. This scheme is defined in [33] as a heterogeneous centralized ensemble, where $\boldsymbol{n}$ FS methods generate $\boldsymbol{n}$ models using the same data.

The main objective of reaching a consensus among several FS methods is to generate a subset of relevant features capable of representing the advantages and disadvantages of all used methods and facing the biases of single methods.

The design built in Figure 2 describes the results of applying the qualitative analysis shown in Figure 1. The process describes a set of data sources to read and analyze to define the concepts related to the topic. Our study selected the concepts by highlighting the relevant concepts that explain implementing the ensemble feature selection. According to phase four of the methodology, the final list of concepts was analyzed to categorize them and identify EFS roles. This phase defines the constraints and special considerations of an EFS method, which are described below:

- Instances must have values in all their columns.
- Instances must not have outliers.
- Values cannot be negative to avoid problems with statistical tests.

Considering the above, datasets must be preprocessed before applying EFS to handle their problems and avoid additional biases in the EFS process. In data mining and machine learning, these constraints are related to preprocessing and preparing data. One of the well-known methodologies to address data mining and machine learning projects is CRISP-DM [34]. The methodology breaks the process of machine learning into six major phases. The data preparation phase describes tasks to cover all the activities required to build the final dataset. One task of the preparation data phase is cleaning data. This task is related to detecting outliers, handling missing values, and fixing the data in a form suitable for the machine learning models. In this sense, the conceptual framework presented in this work only describes a theoretical explanation of ensemble feature selection. The tasks related to phases of data preparation and feature engineering must be addressed in a previous step.

Thus, the ensemble feature selection process is conceived as a machine learning task that needs a proper input dataset. Likewise, considering that real datasets have many problems and need to be preprocessed, the conceptual proposal represents these needs as restrictions that must be solved.

Concepts with a high relation or similarity were grouped or integrated according to phase five. Phase six generated the graphic representation of the framework, which integrated the concepts and relationships previously identified. The authors reviewed and analyzed the model to guarantee the correct representation of the theory extracted from the data sources. Phase seven proposes the validation of the conceptual framework by discussing the proposed model with other researchers. However, the model aims to be a graphic representation of a research topic, but it also aims to validate that the EFS improves the FS process. Thus, its formal validation was carried out as the implementation of a tool to support EFS considering the concepts and relationships of the conceptual framework

3.3. Implementation of the Conceptual Framework

The implementation of the conceptual framework was developed to validate the proposal described in Figure 2. The Scikit-learn Machine Learning Library [35] was used to develop a tool that represents the framework. The solution selects features by different FS algorithms and then aggregates their outputs through consensus. The framework developed allows (i) to read, fix, and impute the values of datasets, (ii) to remove dataset features with high correlation, low variance, or null values, (iii) to generate $\boldsymbol{n}$ subsets of relevant features using n FS algorithms, (iv) to aggregate the subsets generated using methods based on voting, and (v) to evaluate the performance of the subset of features generated by our EFS.

Considering the above, Figure 3 describes the implementation of the framework of Figure 2. The solution groups the functions and methods in packages according to their objective.

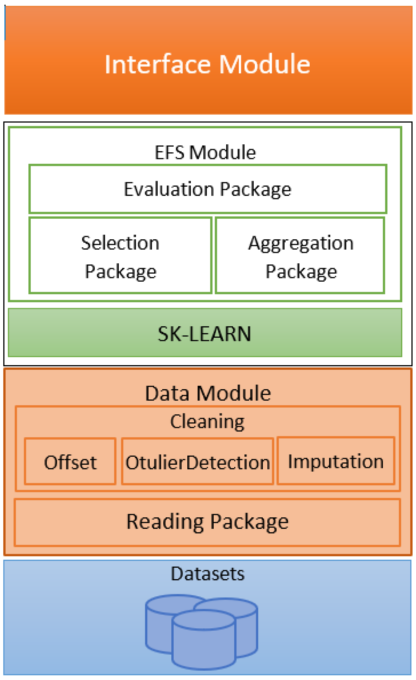


Figure 3. Developed Framework.

- Interface Module: It describes a class. The module exposes the functionalities of the framework to new implementations.
- EFS Module: It is the core of the framework. This component includes all the functionalities associated with the FS based on our ensemble method.

- Evaluation Package: This groups a set of functions to evaluate the accuracy and stability of an EFS output.
- Selection Package: It contains a set of methods to select subsets of relevant features.
- Aggregation Package: It integrates the outputs of n methods of FS using a criterion to aggregate the outputs.
- SCIKIT-LEARN: It is a Machine Learning library that supports the implementation of our EFS package.
- Data Module: It includes the functions to read and preprocess the datasets. The module allows the reading of data from CVS files to adjust them according to assumptions and constraints considered in the conceptual framework design.
- Offset Package: It describes functionalities to calculate an offset dataset to avoid negative values.
- Imputation Package: It implements the Multivariate Imputation to handle datasets with missing values. The authors presented the implementation and evaluation of this component in a previous study [36].
- Outliers Detection Package: This package considers a software component to implement methods of detection of outliers. These methods include, for instance, basic methods as the Standard Deviation or advanced methods as Novelty and Outlier Detection from scikit-learn [37].

It is essential to mention that the constraints defined by the conceptual proposal were solved in the implementation framework with the development of the offset, imputation, and outlier detection packages. However, the framework does not support all data preparation tasks or feature engineering (extraction and transformation of data) with these packages. Therefore, the goal of the packages is only to support common problems associated with the constraints of the FS algorithms.

Additionally, this framework describes generic phases and algorithms that support ensemble feature selection. However, the data science team should adjust hyperparameters for the algorithms considering their data. A detailed example of how to use the EFS framework is described in a previous study developed by the authors to evaluate the EFS quality [38].

3.3.1. Evaluation of the Framework

The evaluation used three public datasets is available on the UCI Machine Learning Repository [20]: Sonar, SPECTF, and WDBC. These datasets were used to compare our results with the EFS algorithm developed in [36]. The results of the evaluation showed the accuracy and stability of the method.

3.3.2. Performance

For the evaluation, the classifiers *Decision Tree Classifier* and *Logistic Regression* used the subsets of features generated by each FS algorithm and the subset generated by the EFS algorithm. Table 1 shows the number of features selected by each FS method.

Table 1. Number of features selected by each FS method.

	Sonar	**SPECTF**	**WDBC**
SelectKBest	5	5	4
RFE	3	3	3
Feature Importance	5	5	4
EFS	10	10	10

To implement the aggregation in the EFS method, the sum of the subsets generated by the n FS algorithms is calculated. Then, for each feature in the subset SUM, an importance index is computed according to Equation (1). The importance of feature i is determined by the number of times it is present in the subset SUM divided by n. Finally, the features that exceed a threshold will be selected in the final set.

$$IF_i = \frac{FF_i}{n} \quad (1)$$

Table 2 compares the results obtained in this study concerning the results obtained in [39] of the feature selection in the datasets Sonar, SPECTF, and WDBC. Column two shows the number of features of each dataset. Columns three and four show the number of features selected by each method, and columns five and six show the percentages of elimination of features obtained in each method. According to the above, the EFS developed in this study was selected for the three datasets subsets with an equal or smaller size than the proposed solution [39]. In this sense, the percentages of elimination of features are equal or higher. To differentiate the results of EFS of [39] and the results of our proposal, we named our framework: F-EFS (framework of ensemble feature selection).

Table 2. Comparison of the EFS method constructed and the results reported in [39].

	Features	**Features Selected by F-EFS**	**Features Selected in [39]**	**% of Elimination of Features by F-EFS**	**% of Elimination of Features by [39]**
Sonar	60	10	24	83.30%	40%
SPECTF	44	10	19	56.70%	43.20%
WDBC	30	10	10	66.70%	66.70%

Tables 3 and 4 show the accuracy obtained by the two classifiers *Logistic Regression* and *Decision Tree Classifier,* when used with each dataset and the subsets generated by FS algorithms and the ensemble method developed.

Table 3. Classification results using Logistics Regression.

	Sonar	**SPECTF**	**WDBC**
SelectKBest	84.05%	53.22%	90.35%
RFE	85.50%	53.22%	93.65%
Feature Importance	84.05%	59.67%	92.10%
F-EFS	86.95%	60.75%	93.85%

Table 4. Classification results using decision trees.

	Sonar	**SPECTF**	**WDBC**
SelectKBest	73.91%	63.44%	89.47%
RFE	65.21%	68.81%	87.71%
Feature Importance	78.26%	72.58%	89.47%
F-EFS	73.91%	74.73%	92.10%

3.3.3. Subsets of Relevant Features

To facilitate the analysis of the results in the evaluation, the features contained in each dataset were named *Fi,* and the target column was called class.

Figure 4 shows the subset of features selected by each algorithm on the Sonar dataset. In this test, the features F9, F10, and F12 were considered relevant by at least two selection algorithms. All algorithms selected F11, F36, and F45, and the other features were selected by only one algorithm. Thus, if the selection threshold defined by the user is 0, the set of features selected would be: {F9, F10, F11, F12, F21, F35, F36, F45, F46, F49}.

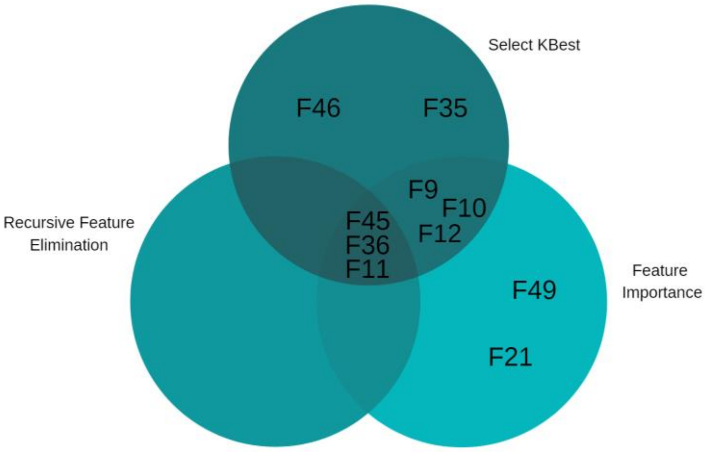


Figure 4. Sonar.

Figure 5 shows the subset of features selected by each algorithm on the SPECTF dataset. The Venn diagram indicates that more than one algorithm chose the features F25, F26, and F40. In comparison, the other features were only considered relevant by only one algorithm. Thus, if we assume that the selection threshold defined by the user is 0, the set of features selected would be {F4, F25, F26, F28, F30, F36, F40, F42, F43, F44}.

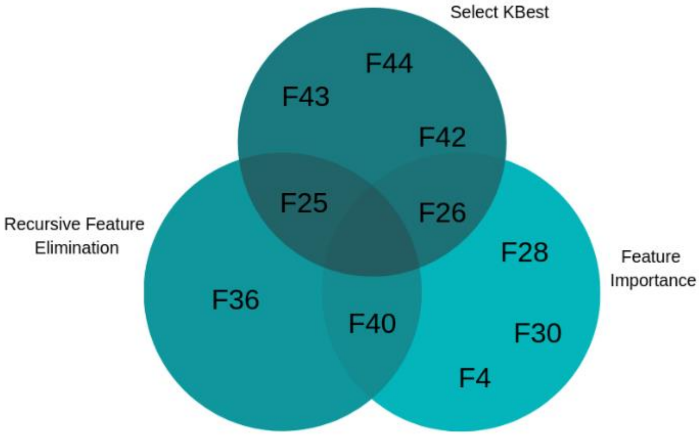


Figure 5. SPECTF.

Figure 6 shows the subsets of features selected by each algorithm for the dataset WDBC. In this result, the features F23 and F24 were chosen by the algorithms Select K Best and Feature Importance, feature F21 by the RFE, and the Feature Importance algorithms, while the rest of the features were selected by only one algorithm. To get all features selected by all FS algorithms, assuming that the threshold defined by the user is 0, the set of features selected would be {F3, F4, F7, F8, F14, F21, F23, F24, F27, F28}.

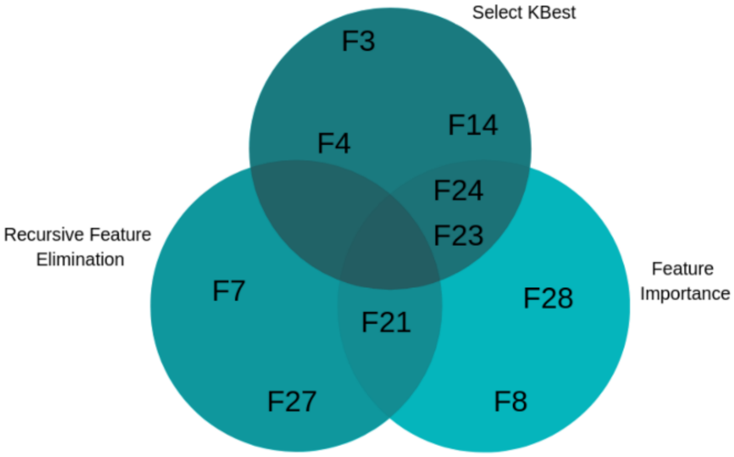


Figure 6. WDBC.

3.3.4. Stability

In solutions based on Ensemble Learning, it is essential to ensure that the outputs of these methods return similar outputs, even if the training data change. This property is known as stability and, according to different studies, there are different measures to evaluate it. The Jaccard index [40] is one of the most common measures for assessing stability in methods that generate subsets of characteristics. The index is described by Equation (2).

$$Jac\ (A, B) = \frac{|A\ \cap\ B|}{|A\ \cup\ B|} \tag{2}$$

To evaluate stability for each dataset, the ensemble method developed was executed 10 times using 10 random samples taken from the original dataset. The results showed that the set generated by the F-EFS was the same in the 10 iterations in the three datasets used.

The results are shown in Table 5.

Table 5. Results of Stability.

	Features	Stability
Sonar	F9, F10, F11, F12, F21, F35, F36, F45, F46, F49	1
SPECTF	F4, F25, F26, F28, F30, F36, F40, F42, F43, F44	1
WDBC	F3, F4, F7, F8, F14, F21, F23, F24, F27, F28	1

4. Discussion

In this study, we proposed the design of a conceptual framework to support ensemble feature selection. Considering a set of concepts and relationships, our proposal explains the general behavior of FS algorithms, their techniques, and how to improve the performance in classification processes. Additionally, the framework provides an overview of existing FS techniques, breaks them down to facilitate understanding, and shows how to combine them to compensate for their biases. Finally, this process allows us to combine outputs of single FS methods and aggregate them by consensus.

Previous studies have proposed solutions based on single FS algorithms focused on a particular domain and problem. For instance, removing features with low variability or identifying relationships among features [41]. However, additional studies have used several FS algorithms to generate a subset of features by each method. These subsets have been tested using classifiers, and the subset with the best performance in the classification is the selected subset [26,42–44]. In the above approach, although they proved different FS algorithms, the final subset of features is influenced by the biases of the algorithm employed to generate it. This is the main difference concerning this study, where the advantages of different methods of FS are considered in the final subset to compensate for their biases.

Besides, recent studies have developed tools to support the combination or assembly of FS algorithms. For example, in [43], the authors proposed a tool developed in R. The tool was designed to combine several outputs of FS algorithms into a random forest algorithm and was used in [39] to select relevant features in the datasets: Sonar, SPECTF, and WDBC. The results in Table 2 showed that our solution achieved an equal or better performance in dimension reduction. However, the comparison was only conducted on dimension reduction. Furthermore, our classification experiments were carried out using random datasets extracted from the original datasets, and the results showed in [39] were not calculated under the same conditions. Consequently, these results cannot be compared.

Likewise, in [28], the author describes a tool developed in Python and available on GitHub. The solution was implemented as a class under the object-oriented programming paradigm, and its methods implement some of the best-known FS algorithms. Each method receives a set of parameters to configure the FS algorithm. The methods can be used as a single FS method, or they can use a particular function to get the subsets selected by several FS algorithms. However, this solution does not provide a mechanism to aggregate all subsets generated.

Considering the studies reviewed and the solution developed, the framework proposed provides an overall scheme to support EFS. Mainly, it facilitates the analysis of techniques, biases, disadvantages, and advantages of the FS algorithms to determine how we can assemble different techniques to get a subset of relevant features more efficiently. In this sense, the framework is designed for data scientists who handle high dimensionality problems and/or few instances.

Regarding the evaluation of our framework, according to the results in Tables 3 and 4, the best classification results were obtained for each dataset using the method developed in this study. For *Sonar* and *WDBC* Table 3 the best performance was achieved using the subsets generated by our method with a logistic regression classifier. The evaluation showed that the accuracy was 86.95% and 93.85%, respectively. On the other hand, for the dataset *SPECTF* the best performance was achieved using our ensemble method with a Decision Tree Classifier with a 74.73% accuracy.

Additionally, the Venn diagrams described in Figures 4–6 show that some features are considered relevant by more than one algorithm. In contrast, others are only present in one. Furthermore, the importance of each feature defined in Formula 1 could also be used as a mechanism based on the weight to provide further relevance in a classification process to the features that are detected with a high differentiation power by several FS algorithms.

Table 5 shows the results of the stability evaluation of the feature sets generated using our ensemble method implemented in the framework. The results show that the method used achieved perfect stability for the three datasets used in the evaluation.

Additionally, it is essential to mention the primary goal of the framework is defining a general approach to support EFS independently of the classification process and datasets. That means data scientists and feature engineers should use our framework to figure out relevant features, and additional tools must be used to build classifiers. As future work, we propose to improve the framework to support the parameterization of new implementations of feature selection methods. Also, to increase the number of consensus methods implemented, offer new measures to evaluate the sets of features selected, and support the ensemble of FS methods in a heterogeneous and homogeneous scheme.

5. Conclusions

A conceptual framework was proposed to clarify the most relevant concepts in the feature selection process. The objective was to define how to improve the performance of the classification algorithms from the sets of features that train them.

The qualitative method followed in order to build the conceptual framework allowed through an exploration of literature to identify concepts and relationships that describe the process of FS and the consensus among different FS techniques.

The conceptual framework built allowed the authors to guide the development of an implementation framework capable of selecting features using an ensemble of FS methods. The chosen process is a set of relevant features with higher performance in classifying the sets of features selected by the single algorithms.

The evaluation allowed us to validate that the performance of a classification process can be optimized by removing irrelevant features. However, the criteria to remove irrelevant features must compensate for the disadvantages of single methods of FS to avoid losing relevant data. Likewise, the performance of our ensemble method achieved 100% stability for the datasets used in the evaluation.

The main contribution of this work to the field of Machine Learning is the definition of a structure that provides an understanding of how to improve the performance of FS based on the consensus of several techniques. This could guarantee better performance in classification algorithms and increase the reliability in those fields of application in which the reliability of the results must be high.

Author Contributions: M.M.-G. conceptualized the idea, developed, and evaluated the proposal, and wrote the original draft of the manuscript. D.M.L. and R.V.-C. reviewed and edited the manuscript, proposed the methodology, and supervised the research. Finally, U.N. verified the results and, reviewed and edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: The work was funded by a grant from the Colombian Agency of Science, Technology, and Innovation Colciencias under Call 647-2015, project "Selection Mechanism of Relevant Features for Automatic Epileptic Seizures Detection." The funder provided support in the form of a scholarship for MMG. Still, the funder did not have any additional role in the study design, data collection, analysis, publishing decision, or manuscript preparation. The specific role of MMG is articulated in the 'Author Contributions' section.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The three datasets used for the evaluation are available on the UCI Machine Learning Repository https://archive.ics.uci.edu/ (accessed on 17 August 2021).

Conflicts of Interest: The authors declare that they have no competing interest. The academic and commercial affiliations of the authors, Colciencias, University of Cauca, and Fraunhofer Center for Applied Research on Supply Chain Services SCS do not alter their adherence Applied Sciences policies on sharing data and materials.

References

1. Pereira, A.G. Selección de Características Para el Reconocimiento de Patrones con Datos de Alta Dimensionalidad en Fusión Nuclear. Ph.D. Thesis, Universidad Nacional de Educacion a Distancia, Bogotá, Colombia, 2015.
2. Guyon, I. An Introduction to Variable and Feature Selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
3. Theodoridis, S.; Koutroumbas, K. *Pattern Recognition*, 2nd ed.; Academic Press: San Diego, CA, USA, 2003.
4. Blum, A.L.; Langley, P. Selection of relevant features and examples in machine learning. *Artif. Intell.* **1997**, *97*, 245–271. [CrossRef]
5. Kohavi, R.; John, H. Artificial Intelligence Wrappers for feature subset selection. *Artif. Intell.* **1997**, *97*, 273–324. [CrossRef]
6. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238. [CrossRef]
7. Sa, N.; Bolo, V. An ensemble of filters and classifiers for microarray data classification. *Pattern Recognit. J.* **2012**, *45*, 531–539.
8. Bolon-Canedo, V.; Sánchez-Maroño, N.; Alonso-Betanzos, A.; Benitez, J.M.; Herrera, F. A review of microarray datasets and applied feature selection methods. *Inf. Sci.* **2014**, *282*, 111–135. [CrossRef]
9. Lee, C.-P.; Leu, Y. A novel hybrid feature selection method for microarray data analysis. *Appl. Soft Comput.* **2011**, *11*, 208–213. [CrossRef]
10. Li, Y.; Wang, G.; Chen, H.; Shi, L.; Qin, L. An Ant Colony Optimization Based Dimension Reduction Method for High-Dimensional Datasets. *J. Bionic Eng.* **2013**, *10*, 231–241. [CrossRef]

11. Cai, R.; Hao, Z.; Yang, X.; Wen, W. An efficient gene selection algorithm based on mutual information. *Neurocomputing* **2009**, *72*, 991–999. [CrossRef]
12. Basto, V.; Yevseyeva, I.; Méndez, J.R.; Zhao, J. A spam filtering multi-objective optimization study covering parsimony maximization and three-way classification. *Appl. Soft Comput. J.* **2017**, *48*, 111–123. [CrossRef]
13. Choi, D.; Ko, B.; Kim, H.; Kim, P. Journal of Network and Computer Applications Text analysis for detecting terrorism-related articles on the web. *J. Netw. Comput. Appl.* **2014**, *38*, 16–21. [CrossRef]
14. den Hartog, D.N.; Kobayashi, V.; Bekers, H.; Kismihók, G. Text Classification for Organizational Researchers: A Tutorial. *Organ. Res. Methods* **2017**, *21*, 1–34.
15. Xia, R.; Xu, F.; Yu, J.; Qi, Y.; Cambria, E. Polarity shift detection, elimination and ensemble: A three-stage model for document-level sentiment analysis. *Inf. Process. Manag.* **2016**, *52*, 36–45. [CrossRef]
16. García-Pablos, A.; Cuadros, M.; Rigau, G. W2VLDA: Almost unsupervised system for Aspect Based Sentiment Analysis. *Expert Syst. Appl.* **2018**, *91*, 127–137. [CrossRef]
17. Bandhakavi, A.; Wiratunga, N.; Padmanabhan, D.; Massie, S. Lexicon based feature extraction for emotion text classification. *Pattern Recognit. Lett.* **2017**, *93*, 133–142. [CrossRef]
18. Mera-Gaona, M.; Vargas-Cañas, R.; Lopez, D.M. Towards a Selection Mechanism of Relevant Features for Automatic Epileptic Seizures Detection. *Stud. Health Technol. Inform.* **2016**, *228*, 722–726.
19. Bolón-canedo, V.; Alonso-betanzos, N.S.A. Feature selection for high-dimensional data. *Prog. Artif. Intell.* **2016**, *5*, 65–75. [CrossRef]
20. Dheeru, D.; Taniskidou, E.K. *UCI Machine Learning Repository*; University of California, Irvine, School of Information and Computer Sciences: Irvine, CA, USA, 2017.
21. Chang, C.; Lin, C. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–39. [CrossRef]
22. Bay, S.D. Combining Nearest Neighbor Classifiers Through Multiple Feature Subsets. In Proceedings of the Fifteenth International Conference on Machine Learning, Madison, WI, USA, 24–27 July 1998; pp. 37–45. Available online: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.114.4233&rep=rep1&type=pdf (accessed on 17 August 2021).
23. Zheng, Z.; Webb, G.I.; Ting, K.M. Integrating boosting and stochastic attribute selection committees for further improving the performance of decision tree learning. In Proceedings of the Tenth IEEE International Conference on Tools with Artificial Intelligence (Cat. No.98CH36294), Taipei, Taiwan, 10–12 November 1998; pp. 321–332. Available online: https://ieeexplore.ieee.org/document/744846 (accessed on 17 August 2021). [CrossRef]
24. Opitz, D.W. Feature Selection for Ensembles. In *National Conference on Artifi.cial Intelligence*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 379–384. Available online: https://www.aaai.org/Papers/AAAI/1999/AAAI99-055.pdf (accessed on 17 August 2021).
25. Piao, Y.; Piao, M.; Park, K.; Ryu, K.H. An ensemble correlation-based gene selection algorithm for cancer classification with gene expression data. *Bioinformatics* **2012**, *28*, 3306–3315. [CrossRef]
26. Mohammad, L.; Tajudin, A.; Al-betar, M.A.; Ahmad, O. Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering. *Expert Syst. Appl.* **2017**, *84*, 24–36. Available online: https://ur.booksc.eu/book/67787096/455350 (accessed on 17 August 2021).
27. Neuman, U.; Genze, N.; Heider, D. EFS: An ensemble feature selection tool implemented as R-package and web-application. *BioData Min.* **2017**, 1–9. Available online: https://biodatamining.biomedcentral.com/articles/10.1186/s13040-017-0142-8 (accessed on 17 August 2021).
28. Koehrsen, W. A Feature Selection Tool for Machine Learning in Python, Towards Data Science. 2018. Available online: https://towardsdatascience.com/a-feature-selection-tool-for-machine-learning-in-python-b64dd23710f0 (accessed on 7 November 2018).
29. Jabareen, Y. Building a Conceptual Framework: Philosophy, Definitions, and Procedure. *Int. J. Qual. Methods* **2009**, *8*, 49–62. [CrossRef]
30. Liu, H.; Motoda, H. *Feature Selection for Knowledge Discovery and Data Mining*; Springer: Berlin/Heidelberg, Germany, 1998.
31. Kuncheva, L.I. *Combining Pattern Classifiers: Methods and Algorithms*; Wiley-Interscience: Hoboken, NJ, USA, 2004. Available online: https://www.springer.com/gp/book/9780792381983 (accessed on 17 August 2021).
32. Yu, L.; Liu, H. Efficient Feature Selection via Analysis of Relevance and Redundancy. *J. Mach. Learn. Res.* **2004**, *5*, 1205–1224.
33. Seijo-Pardo, B.; Porto-Díaz, I.; Bolón-Canedo, V.; Alonso-Betanzos, A. Ensemble feature selection: Homogeneous and heterogeneous approaches. *Knowl.-Based Syst.* **2017**, *118*, 124–139. [CrossRef]
34. IBM. *Manual CRISP-DM de IBM SPSS Modeler*; IBM Corp.: Armonk, NY, USA, 2012; p. 56. Available online: https://www.ibm.com/docs/es/spss-modeler/SaaS?topic=guide-introduction-crisp-dm (accessed on 17 August 2021).
35. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
36. Mera-Gaona, M.; Neumann, U.; Vargas-Canas, R.; López, D.M. Evaluating the impact of multivariate imputation by MICE in feature selection. *PLoS ONE* **2021**, *16*, e0254720. [CrossRef] [PubMed]
37. Scikit-Learn. Documentation—Scikit-Learn. 2021. Available online: https://scikit-learn.org/stable/modules/outlier_detection.html (accessed on 16 August 2021).
38. Mera-Gaona, M.; López, D.M.; Vargas-Canas, R. An Ensemble Feature Selection Approach to Identify Relevant Features from EEG Signals. *Appl. Sci.* **2021**, *11*, 6983. [CrossRef]

39. Neumann, U.; Riemenschneider, M.; Sowa, J.-P.; Baars, T.; Kälsch, J.; Canbay, A.; Heider, D. Compensation of feature selection biases accompanied with improved predictive performance for binary classification by using a novel ensemble feature selection approach. *BioData Min.* **2016**, *9*, 1–14. [CrossRef]
40. Kalousis, A.; Prados, J.; Hilario, M. Stability of feature selection algorithms: A study on high-dimensional spaces. *Knowl. Inf. Syst.* **2007**, *12*, 95–116. [CrossRef]
41. Lachner-Piza, D.; Epitashvili, N.; Schulze-Bonhage, A.; Stieglitz, T.; Jacobs, J.; Dümpelmann, M. A single channel sleep-spindle detector based on multivariate classification of EEG epochs: MUSSDET. *J. Neurosci. Methods* **2018**, *297*, 31–43. [CrossRef]
42. Su, J.; Yi, D.; Liu, C.; Guo, L.; Chen, W.-H. Dimension Reduction Aided Hyperspectral Image Classification with a Small-sized Training Dataset: Experimental Comparisons. *Sensors* **2017**, *17*, 2726. [CrossRef] [PubMed]
43. Khair, N.M.; Hariharan, M.; Yaacob, S.; Basah, S.N. Locality sensitivity discriminant analysis-based feature ranking of human emotion actions recognition. *J. Phys. Ther. Sci.* **2015**, *27*, 2649–2653. [CrossRef] [PubMed]
44. Garbarine, E.; DePasquale, J.; Gadia, V.; Polikar, R.; Rosen, G. Information-theoretic approaches to SVM feature selection for metagenome read classification. *Comput. Biol. Chem.* **2011**, *35*, 199–209. [CrossRef] [PubMed]