Ensemble Learning Based Feature Selection Using Convex Concave Programming

Pınar Karadayı Ataş

Faculty of Engineering and Natural Sciences,
Department of Computer Engineering, Bahçeşehir University, Istanbul, Turkey

Süreyya Özöğür - Akyüz

Faculty of Engineering and Natural Sciences,
Department of Mathematics, Bahçeşehir University, Istanbul, Turkey

Abstract

Ensemble feature selection and multiple classifier systems have recently gained importance in machine learning. Ensemble learning improves learning ability by combining several models, an improvement which leads to better predictive performance than a single model. In recent years, ensemble based feature selection approaches have been proposed in which, multiple diverse feature selection methods are combined. These approaches are superior to traditional feature selection techniques in various aspects. In this paper, we propose a novel ensemble based feature selection algorithm using Convex Concave Programming, which is based on ensemble pruning. The optimization model in the pruning step selects the best subset of the ensemble, simultaneously considering the models' accuracy and diversity. The proposed algorithm was tested on multiple data sets and learning performances are compared with various feature selection algorithms. The empirical results shows that the proposed algorithm performs at higher classification accuracy.

Keywords: Feature selection, Ensemble Learning, Ensemble Pruning, Dynamic Ensemble Selection(DES), Convex Concave Programming

1. Introduction

Feature Selection is the process of choosing the most relevant and important features which contribute to learning with the highest prediction accuracy. Feature selection methods have various applications [1, 2], the determination of which constitutes the data pre-processing step of machine learning problems. It is important to eliminate irrelevant features which do not have any dependency on the target value since those features reduce the prediction accuracy of the learning model. There exist many feature selection methods in the literature, including filters based on distinct metrics like probability, entropy, information theory, embedded, and wrapper methods, all using different algorithms [3]. Most feature selection methods are wrapper methods, which evaluate the features using the learning algorithm. Algorithms based on the filter model examine the intrinsic properties of the data to evaluate the features before the learning tasks. Filter-based approaches almost always rely on class labels, most commonly assessing correlations between features and class label. Some typical filter methods include data variance, Pearson correlation coefficients, Fisher score, and the Kolmogorov-Smirnov test.

Ensemble based feature selection methods are designed to generate an optimum subset of features by combining multiple feature selectors based on the intuition behind the ensemble learning. The general idea of ensemble feature selection is to aggregate the decisions of diverse feature selection algorithms to improve representation ability. Recent studies show that the decision of an ensemble of feature selection algorithms gives more accurate prediction than any single feature selection technique [4, 5].

Ensemble based feature selection methods involve two major steps: generation of diverse feature selectors and aggregation of the decisions. There are three types of generation approaches studied in the literature employed to construct a diverse ensemble library, and which can be listed as follows:

1. Data Variation Methods,
2. Function Variation Methods,

3. Hybrid Variation Methods.

The first approach, *Data Variation*, creates subsets of samples by using different methods, such as bagging [6] or boosting[7] or using different feature subspaces and random subspaces [8]. In the second method, *Function Variation*, the diversity of an ensemble is provided by the diversity of feature selection functions. Here, the most common functions are filter based rather than wrapper approaches because of their advantages in computational cost. Unlike the first two methods, *Hybrid Variation Methods* aggregate both data variation and function variation steps since it is argued that including data variation or function variation alone is not enough to create a robust ensemble [9]. In [10],the similarity between the function variation and hybrid variation is higher than the similarity between the data variation. Furthermore, function and hybrid variation methods produce higher classification performance than data variation.

In the literature, there are two main approaches regarding the use of ensembles in feature selection. In the first, feature selection steps are used for obtaining the diversity needed for using posterior ensemble classification methods[11]. Other authors use ensembles of feature selectors to improve the accuracy, diversity, and stability of the feature selection process [12, 13, 14, 15]. This latter approach is of special interest in knowledge discovery scenarios, and mainly in high dimensional cases.

In [16], five different pairwise measures of diversity were compared over 21 datasets with fixed ensemble sizes. They aimed to design a fitness function that shows the relation between accuracy and diversity. The results showed that there is a close relationship between the functions employed and the number of ensemble members that produce the highest accuracy. Other works, such as Bolón-Canedo et al. [17] used a fixed number of filters in high dimensional scenarios. In [18] two different basic methods of heterogeneous type were proposed. The first method applied five filters that fed five classifiers followed by the aggregation step. The same filters were used in the second formulation with the aggregation step, previous to classification. In [19], a new algorithm

named Multicriterion Fusion-based Recursive Feature Elimination was developed. Its aim is to increase the robustness of feature selection algorithms by using multiple feature selection evaluation criteria. Another study used Multilayer perceptrons at the ensemble stage [20].

Diversity is a factor that deserves specific emphasis. By using several types of feature selectors in an ensemble [3] such as rankers, subset methods filters, wrappers, embedded methods, or univariate and multivariate methods as in [13, 18] we can provide diversity.

In [21], several ensembles of filter rankers were applied to the area of software quality. The combination of individual rankings included simple methods like mean, median, and minimum, and complex methods such as Complete Linear Aggregation [22] (CLA), Robust Ensemble Feature Selection (Rob-EFS) [23], SVM-Rank [13], and data complexity measures [24]. Meanwhile, there exist many parallel and distributed implementations of feature selection methods [25, 26]Further, various research projects have developed ensembles making use of distributed or parallel schemes. In [13]a heterogeneous approach was proposed, with the idea of distributing the dataset in several nodes, applying the same feature selection method in each of them, and then at the end of their work aggregating the results. Hong et al [27].also developed a feature selection algorithm for unsupervised clustering which put together the clustering ensemble method and the population-based incremental learning algorithm.

The same authors also developed the task of feature ranking for unsupervised clustering [28] for guiding computation of features' relevance. A different approach was followed in the work by Morita et al. [29], in which they developed an ensemble of classifiers based on unsupervised feature selection. Bellal et al. [30] developed a new method called semi-supervised ensemble learning guided feature ranking method (SFR), which combined a bagged ensemble of standard semi-supervised approaches with permutation-based out-of-bag feature importance. A new wrapper-type semi-supervised feature selection framework which finds the relevant features using confident unlabeled data was developed by Han et al. [31]. They employed an ensemble classifier that supports the estimation

of confidence in unlabeled data. Ko et al. [32] developed a dynamic classifier selection approach where the competencies of the individual classifiers are calculated during the classification step. In their study, majority voting is used rather than the static selection method because of its superior performance. In this paper, DES was used as classifier method in all classification steps in the proposed Algorithm 1.

It must be noted that in each of the above methods, the number of functions, i.e., the cardinality of an ensemble library, is not determined theoretically. The number of functions in the ensemble acts as a hyper-parameter of these methods which directly affects the classification performance in the aggregation step. In the generation step of the ensemble, the most accurate and diverse models are desired for better prediction performance at the end. However, there might be models which are weak in the generation phase, causing a decrease in overall accuracy. To eliminate such redundancies in the ensemble, a pruning step is needed to select the optimum subset of the ensemble. To the best of our knowledge, no pruning algorithm has been proposed for ensemble-based feature selection algorithms. There exist pruning methods developed for ensemble classification of multi-class task problems using Error-Correcting Output Codes [4, 5]. In these studies, the importance of the accuracy and diversity trade-off is highly emphasized and optimization-based approaches are proposed for ensemble classification models. This trade-off can be explained as follows: High accuracy in the ensemble leads to a decrease in the diversity of the ensemble, and an increase in diversity sacrifices the accuracy of the overall ensemble.

In this study, we propose a novel ensemble based feature selection algorithm that fills the gap in the literature regarding feature selection problems, described above, by using an optimization model to simultaneously optimizing the accuracy and diversity trade-off. Since the pruning step of the proposed approach here involves an optimization model, the cardinality of the subset of an ensemble is not a hyper-parameter anymore, as it is obtained directly as a solution of the optimization model. The rest of this paper is structured as follows: In Section 2,background methods are provided and in Section 3, various traditional feature

selection algorithms that generate the ensemble are summarized. The proposed ensemble feature selection technique is presented in 4 with the experiment and results in Section 5. We conclude with some final remarks and ideas for future work in Section 6.

2. Background Material

In this section, various background methods used in the pruning, classification steps of this study are introduced. In the following subsection, Disciplined Convex-Concave Programming (DCCP), the core of the pruning step, is introduced briefly. In the later subsection, the classifier in this study, Dynamic Ensemble Selection(DES), is summarized.

2.1. Disciplined Convex-Concave Programming

DCCP is an optimization method which was first introduced in [33] and which combines two ideas: Disciplined Convex Programming (DCP) and Convex-Concave Programming (CCP) [34]. DCP requires a set of conventions in which problems follow, whereas CCP is an organized heuristic for solving nonconvex problems. Disciplined convex programming can be defined by the following optimization problem:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f_0(x) - g_0(x) \\ \text{subject to} \quad & f_i(x) - g_i(x) \leq 0,\ i = 1, \ldots, m, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$ refers to the optimization variable, and the functions $f_i : \mathbb{R}^n \to \mathbb{R}$ and $g_i : \mathbb{R}^n \to \mathbb{R}$ $(i = 1, ..., m)$ are convex functions.

Above problem (1) can be rewritten as follows:

$$\begin{aligned} \text{minimize} \quad & f_0(x) - t \\ \text{subject to} \quad & t = g_0(x), \\ & f_i(x) \leq g_i(x),\ i = 1, \ldots, m, \end{aligned} \tag{2}$$

where x and t refers to the original optimization variable and a new optimization variable, respectively.

DCCP is an appropriate and simple standard form for Difference Programming (DC), because the linearized problem of CCP is a DCP problem if the original problem is DCCP. The linearized problem can then be transformed into a cone program and solved using generic solvers [34].

2.2. Dynamic Ensemble Selection - COMBO

In this study, Dynamic Ensemble Selection(DES) was used as a classifier for the proposed ensemble model. As previously reported, given that selecting only one classifier is very vulnerable to error, some researchers chose to pick a subset of classifiers. Ko et al. [32] suggested an approach aimed at imitating the Oracle model, which obtained the best ensemble results [35]. The KNORA-E (K Nearest ORAcles - Eliminate) eliminates the classifier from the ensemble if the classifier misclassifies any pattern of the neighbors. There exists moreover a weighted form KNORA-E-W which weights the labels of the chosen classifiers based on the distance between the test sample and the neighbors. This work includes two fusion algorithms: KNORAU (K Nearest ORAcles - Union) and its weighted form KNORA-U-W. Soares et al. [36] choose the N most correct classifier, according to a defined region of competence, and the J most diverse classifiers in order to produce the ensemble. The values of N and J were settled by the authors. These methods are called dynamic ensemble selection (DES) since they can choose more than one classifier. In this paper, DecisionTreeClassifier, LogisticRegression, KNeighborsClassifier, RandomForestClassifier,, GradientBoostingClassifier were set as estimators of classifer steps. Here, we used Combo which is a relatively new library specialized in ensemble learning which provides several common methods under a unified Scikit-learn-compatible API so that it maintains compatibility with many estimators from the Scikit-learn ecosystem [37]. The Combo library delivers algorithms that are capable of combining models for classification, clustering, and anomaly detection tasks, and it has been used widely in the Kaggle predictive modeling community. Combo

provides a unified outlook for different ensemble methods whilst remaining compatible with Scikit-learn.

2.3. Joint Criterion Method

In the Joint Criterion Method, quality and diversity terms are merged into a joint criterion function [38]. For a given ensemble size K, the following objective function (3) is maximized with respect to find the indices of the best candidates among the ensemble

$$\alpha \sum_{i=1,\ldots,K} SNMI(C_i, L) + (1-\alpha) \sum_{i \neq j} (1 - NMI(C_i, C_j)), \tag{3}$$

where the first term measures the quality, the second term measures the diversity and the parameter α controls the impact assigned to each term [38]. It starts with a single solution having the highest-quality and the next candidate is added to the ensemble subsequently which maximizes the objective function of the problem (3) [38].

3. Generation of Ensemble Library

In this study, 28 different traditional feature selection algorithms are used to create the ensemble of feature selection methods. These are grouped into four categories similarity based, information theoretical based, sparse learning based and statistical based methods.

3.1. Similarity Based Methods

In general, feature selection algorithms utilize a variety of criteria distance, separability, information, correlation, dependency, and reconfiguration error to define attribute appropriateness. Similarity-based feature selection methods assess the importance of preserving data similarity and the importance of features. They are divided into five sub-categories as follows:

1. **Laplacian Score**

   The Laplacian Score (LS) is an uncontrolled and three-phase attribute selection algorithm that can best protect the data manifold structure [39]. It

is generated by Laplacian Eigenmaps [40] and Locality Preserving Projection [41] which evaluates the features according to their locality preserving power.

2. **Spectral Feature Selection (SPEC)**

   SPEC is a graph based feature selection method which is an extension of LS. It can be used for both supervised and unsupervised scenarios. For example, in the unsupervised case, Radial Basis Function (RBF) kernel function is used to measure data similarity. In the supervised case, a diagonal matrix is constructed using affinity matrix information [42].

3. **Fisher Score**

   Fisher score is supervised feature selection methods that selects each feature independently according to their scores under the Fisher criterion [43].

4. **Trace Ratio Criterion**

   In the Trace Ratio Criterion method, a feature subset is selected based on the corresponding subset-level score, which is calculated in a trace ratio form [44].

5. **ReliefF**

   ReliefF algorithm is one of the most successful filtering feature selection methods. It selects features to separate instances from different classes [45]. It assesses the quality of features based on how well their values discriminate between samples that are near each other.

3.2. Information Theoretical Based Methods

Information theoretical based methods use different heuristic filter criteria to measure the importance of the attributes which maximize the relevance of the attributes and minimize their redundancy [43]. These types of methods can be divided into nine sub-categories as follows:

1. **Mutual Information Maximization (MIM) (or Information Gain)**

   MIM evaluates the significance of a feature by its correlation with the class label [46].

2. **Mutual Information Feature Selection (MIFS**)

The MIFS criterion considers both feature relevance and feature redundancy in the feature selection phase [47].

3. **Minimum Redundancy Maximum Relevance (MRMR)**

The MRMR criterion considers both feature with maximum relevance and feature with minimum redundancy in the feature selection phase [48].

4. **Conditional Infomax Feature Extraction (CIFE)**

As long as the feature redundancy of a given class label is stronger than the intra feature redundancy, the feature selection is affected negatively [49]. CIFES takes this into account by including a third term which maximizes the conditional redundancy between unselected features and already selected features for a given class label.

5. **Joint Mutual Information (JMI)**

MIFS and MRMR reduce feature redundancy in the feature selection process. It is recommended that JMI, an alternative criterion, increase the shared information between the new selected attribute, and the selected attributes given the class labels [50]. The basic idea of JMI consists of adding new features that are complementary to existing features for a given class label.

6. **Conditional Mutual Information Maximization (CMIM**)

CMIM selects features iteratively by maximizing the mutual information with the class labels given the selected features [51, 52].

7. **Double Input Symmetrical Relevance (DISR)**

DISR performs normalization techniques to normalize mutual information [53].

8. **Fast Correlation Based Filter (FCBF)**

FCBF is an algorithm that has the capability of being employed as the approximation method for relevance and redundancy analysis [54].

9. **Interaction Capping**

The Interaction Capping feature selection criterion is similar to CMIM except that Interaction Capping restricts the term $I(X_j; X_k) - I(X_j; X_k \,|Y\,)$

to be nonnegative where $I(.,.)$ refers to the information gain function [55].

3.3. Sparse Learning Based Methods

Filter based feature selection methods select attributes that are independent of any learning algorithm. The bias of the learning algorithm is not taken into account in filter type approaches, so that the selected attributes may not be optimal for a specific problem. In order to overcome this issue, embedded type approaches are developed which embed the feature selection step into the learning model construction so that each step feeds into one other. There are three types of embedded feature selection methods: The first is based on pruning redundant features by assigning binary weights to features while maintaining prediction accuracy. The second type consists of a built-in feature selection mechanism such as ID3 [56] and C4.5 [57]. The last type refers to sparse learning based methods, which minimize empirical error by inducing a regu- larization term to the objective function so that some feature coefficients are small or exactly zero. There are different types of sparse based approaches, but we will introduce only those that are used in this study.

1. **Multi-Cluster Feature Selection (MCFS)**

   Most of the existing sparse feature selection methods use label information of the data where the feature selection step is modeled after determining the sparse feature coefficients. Since labeled data is costly and time consuming to collect, unsupervised sparse learning based feature selection has gained increasing attention in recent years [58, 59]. MCFS is one of the first unsupervised feature selection algorithms developed and performs spectral clustering and sparse coefficient learning before the feature selection step [60].

2. **Feature Selection with L_1 norm Regularization**

   This method performs feature selection by assigning insignificant input features with zero weight and useful features with a non zero weight by incorporating l_1 norm penalty functions to the objective function while minimizing the empirical error on the training set [61].

3. $l_{2,1}$ **norm Regularized Discriminative Feature Selection**

A widely accepted criterion for choosing an unsupervised feature is to select attributes that best protect the manifold structure of the data [39]. One crucial property of 2,1-norm regularization is that it allows multiple predictors to share similar sparsity patterns. However, the resulting optimization problem is difficult to solve because of the non-smoothness of 2,1-norm regularization.

4. **Nonnegative Discriminative Feature Selection (NDFS)**

NDFS is an algorithm that performs spectral clustering and attribute selection at the same time to select a subset of distinctive attributes. [62]. Unlike other spectral clustering methods, NDFS executes nonnegative and orthogonal constraints in the spectral clustering phase which causes the learned pseudo class labels to be closer to real cluster results.

3.4. Statistical based Methods

Another feature selection algorithm category is based on various statistical measurements. Because they rely on statistical criteria instead of learning the algorithm to assess the appropriateness of attributes, most of these methods are filter-based methods. We can divide statistical methods into three categories:

1. **F-score**

In statistical analysis of binary classification, F-score is a measure of a test's accuracy. It considers both the precision and the recall of the test to obtain the scores [63].

2. **Gini Index**

Gini index is a statistical measure to quantify if the feature is able to separate instances from different classes [64].

3. **Correlation Based Feature Selection (CFS)**

The basic idea of CFS is a heuristic approach based on a correlation to evaluate the value of the attribute subset [65].

3.5. Feature Selection with Structure Features

Most of the feature selection algorithms are based on the assumption that the features are independent from each other though the essential structures among them are disregarded. Yet, in many real problems, features reveal various types of structures such as spatial or temporal smoothness, disjointed groups, trees and graphs [66]. Next, we briefly give the idea of graph based and group based approaches.

1. **Feature Selection with Graph Feature Structures**

   In many cases, strong dependencies may occur between the attributes so that an unverified graph can be used to encode these dependencies such that nodes represent features and edges between two nodes showing the pairwise dependencies between features [60]. Those dependencies on the graph can be transformed to a more mathematical representation by adjacency matrices consisting of binary entries.

2. **Feature Selection with Group Feature Structures**

   In many real-world applications, features represent group structures. One of the most common examples is seen in multi-factor analysis-of-variance (ANOVA), where each factor is associated with several groups. When selecting attributes, this method obtains accurate predictions when the group structure between attributes is considered [60].

3.6. Wraper Methods

Wrapper methods consider the selection of a set of features as a search problem, where different combinations are prepared, evaluated, and compared to other combinations [67, 68].

4. Proposed Mathematical Model

In this study, we developed a model which determines the optimum subset of the different solutions among a library of 28 feature selection methods (summarized in the previous sections). After the generation step of the ensemble of

feature selectors on the training set, their accuracies and diversties are calculated by using dynamic ensemble selection which form the entries of matrix T defined by equation 4 below:

$$[T] = \begin{cases} Acc_i, & i = j \\ \sum\{Y_i^{\mathrm{DES}} \neq Y_j^{\mathrm{DES}}\}, & i \neq j \end{cases} \tag{4}$$

In the matrix T, the total number of correct predictions of the i-th feature selector by classifier DES, represented by Acc_i are defined on the diagonal entries and the total number of uncommon predictions of these feature selectors are assigned to the off-diagonal entries as a measure of diversity in T. In this way, diagonal elements of the matrix T represent the accuracy criterion whereas the off-diagonal elements represent diversity.

Here, we adapted our previous study for *ensemble clustering selection* approach in [69] and [70] to the *ensemble based feature selection model*. The previous model differs by involving different accuracy and diversity metrics in the matrix T. In [69] and [70], since the problem was a clustering problem which did not have label information, normalized mutual information values were previously used to define accuracy and diversity metrics.

In this paper, quality (accuracy) and diversity metrics are defined to be the total number of accurate predictions and the uncommon pairwise predictions of each binary couple of feature selectors, respectively. Basically, the main intuition behind maximizing the accuracy and diversity trade off within the ensemble learning library studied previously in [4, 5, 69, 70] is to adapt the ensemble of feature selectors with the same optimization model (5) by adapting the accuracy diversity notion using metrics based on feature selectors as follows:

$$\begin{aligned} &\text{minimize } x^T T x \\ &\text{subject to } \textstyle\sum_{i=1}^{n} x_i = k, \\ &x_i \in \{0,1\}\ (i = 1, 2, \ldots, n), \end{aligned} \tag{5}$$

where k stands for the pruning rate of the ensemble, i.e., the cardinality of the

subset of the ensemble. Since above 0- 1 binary integer problem (5) is NP-hard in general. However, there exist studies that approximates the solution of integer programming optimally in the literature for many years. One of the recent study in the field of machine learning, presents a new formulation of the classical univariate decision tree problem as an Mixed Integer Programming (MIP) problem that motivates their new classification method which is called Optimal Classification Trees (OCT) [71]. In the of the applications of MIP involves with the tensor complementarity problem where a global solver LINGO was used to obtain optimal solution [72]. The proposed ensemble pruning model for clustering problem in this paper was inspired from the MIP in [5] which includes the parameter k in its constraint. The constraint in problem (5) determines the size of the subset of the ensemble, in other words, the parameter k is given by the user beforehand as a pruning rate. Furthermore, its variables are integer because it is defined with a sum that counts the number of elements to be selected in the new subset of ensemble which can be defined as cardinality constraint by a zero norm. Thus, the solution of the MIP and the accuracy of the machine learning model highly depend on the optimal value of parameter k. The objective of our study is to get rid of the parameter k to automate finding the optimal value of k while selecting the best candidates considering both accuracy and diversity within the optimization problem. In order to do this, we moved that cardinality constraint to the objective function with a regularization constant so that the whole MIP turned into continuous optimization problem. This procedure can also be regarded as regularization in statistical learning to overcome the complexity as in Lasso regularization

This relaxation by moving the cardinality constraint to the objective function with a regularization constant ρ is further improved by adding bound constraint to variable x to obtain sparse solution as shown below:

$$\begin{aligned} \underset{x\in\mathbb{R}^n}{\text{minimize}} \quad & x^T T x + \rho\|x\|_0 \\ \text{subject to} \quad & x^T x = 1, \end{aligned} \tag{6}$$

Since the model (5) is relaxed to a continuous programming, in order to keep

the sparsity, an additional constraint that bounds x is added to the problem (6).

The proposed ensemble based feature selection model introduced by equation (6) is non-convex because of the second term and the matrix T might be negative definite. Approximating the zero norm with the student log likelihood distribution and adding/subtracting the term τI to the first term leads to a difference of convex functions where τ is defined to be $\tau \geq \max\{0, -\lambda_{min}(T)\}$. Hence the optimization problem (6) can be rewritten as:

$$\begin{aligned} \underset{x\in\mathbb{R}^n}{\text{minimize}} \quad & \left\{ \tau \left\|x\right\|_2^2 - \left[x^T \left(\tilde{T} + \tau I \right) x - \rho \sum_{i=1}^{n} \frac{\log\left(1 + \left|x_i\right| / \epsilon\right)}{\log\left(1 + 1/\epsilon\right)} \right] \right\} \\ \text{subject to} \quad & x^T x = 1, \end{aligned} \tag{7}$$

where $\hat{T}$ is the normalized form of the matrix T and ρ refers to the regularization parameter corresponding to the cardinality of a subset of the ensemble which is introduced by a zero norm.

If the absolute value in equation (7) is replaced with an additional variable y_i by adding an extra constraint $-y \leq x \leq y$, the model (7) takes the following final form:

$$\begin{aligned} \underset{x,y\in\mathbb{R}^n}{\text{minimize}} \quad & \left\{ \tau \left\|x\right\|_2^2 - \left[x^T \left(\tilde{T} + \tau I \right) x - \rho \sum_{i=1}^{n} \frac{\log\left(1 + y_i/\epsilon\right)}{\log\left(1 + 1/\epsilon\right)} \right] \right\} \\ \text{subject to} \quad & -y \leq x \leq y, \\ & x^T x = 1. \end{aligned} \tag{8}$$

5. Experiments and Results

In this work, the five most popular feature selection data sets, selected from different domains are used [73]. The features that exist in these datasets are either numerical or categorical values. The number of features, number of instances and number of classes are presented in Table (1).

Dataset	Type	Feature Value	# Feature	# Instance	# Class
Lung small	Bio	Discrete	325	73	7
Madelon	Artificial	Continuous	500	2600	2
Yale	Image	Continuous	1024	165	15
WarpAR10P	Image	Continuous	2400	130	10
Colon	Bio	Discrete	2000	62	2
Urban Land Cover	Physical	Continuous	148	168	9
Libras Image	Bio	Continuous	91	360	15
Hill-Valley	Physical	Continuous	101	606	2

Table 1: Detailed information of benchmark datasets.

The dataset is divided into three parts:testing, validation, and training. 20% of the entire dataset is used for testing, and remaining 80% is divided into (20%) validation and (80%) training folds. 28 different feature selection algorithms were applied on training data. DES were employed with 5-fold cross-validation with the selected features by the 28 different feature selection techniques in order to build T matrix with accuracy and diversity entries.

Optimization problem (8) is solved by the DCCP algorithm [33] of cvxpy library of Python 3.7. The solution of the optimization model defined by equation (8) provides the indices of the selection of the results of the 28 attribute selection methods. The hyper-parameter ρ was determined by 5 fold cross-validation on the validation set among the interval of $\rho = \left[10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^{2}, 10^{3}\right]$ and the threshold value of 0.01 was determined experimentally for x values to binarize it.The coordinates of vector x represent the feature selection methods and the indices of vector x having 0 values corresponds to redundant feature selectors being eliminated whereas indices of values of 1 correspond to the methods which are to be chosen.

The solution of the DCCP model corresponding to the optimum ρ parameter refers to the best subset among 28 feature selection methods. The elements of this subset are composed of the different attribute sets S_i^f introduced in

Algorithm 1 that are generated by the those feature selection methods outputted by the DCCP method. The voting algorithm, was applied to aggregate the results of subsets S_i^f of the feature selectors. In this voting step, the solutions of the algorithms that were voted more than 50% were included in the selected attributes. We used voting method to select features as an aggregation of feature selector in the final subset determined by DCCP. After voting step (best features are selected) classification step was performed by using DES. Furthermore, final classification is also performed by DES. In full ensemble there may be methods which decrease the perfomance of the ensemble. Our main goal is to eliminate such methods by pruning. One can increase the number of possible method candidates in the ensemble so that diversity increases. All pruning methods compared including the proposed DCCP model and also full ensemble results find the final set of features using voting. In this manner, voting should be considered as an aggregation function for both pruned and unpruned cases.

For example, if the solutions of 15 methods from 28 methods were selected to the subset by using the DCCP model (8), the attributes of those 15 techniques which passed the 50% of threshold would be considered to be the final attributes of the test data. All the steps described here are given in the flowchart shown by Algorithm 1 and Figure 1 and the performance of the models was compared with the methods in the literature called Joint Criterion [38].

Algorithm 1 Improved Ensemble Feature Selection with DCCP

Input: $X^{tr},\ X^{val},\ X^{test},\ S_n$ (Feature Selection Algorithms)

Parameter: ρ (DCCP parameter), k (Number of Features)

Output: percentage of accuracy

1: **for** $i \leftarrow 1$ to n **do**

2: $S_i^f = S_n(X)$ /* the feature selection subset that each feature selection algorithm chooses */

3: $Acc_i = acc(\mathrm{DES}(X^{tr}_{S_i^f}))$ /* DES percentage of accuracies using the training set */

4: **end for**

5: $T_{ii} = Acc_i$ /* The percentage of accuracies for feature selection methods/*

6: $T_{ij} = \sum_{i \neq j} Y_i^{\mathrm{DES}} \neq Y_j^{\mathrm{DES}}$ /* non-common estimation results for i-th and j-th feature selection methods */

7: $\breve{\mathrm{S}}_\rho = \mathrm{DCCP}(T, \rho)$ /* Obtaining the optimum subset of feature selection algorithms with DCCP method on X^{val} */

8: $F = Votting(\breve{\mathrm{S}}_\rho)$ on X^{test}

9: $Percentage\ of\ Accuracy = \mathrm{DES}(X^{test})$

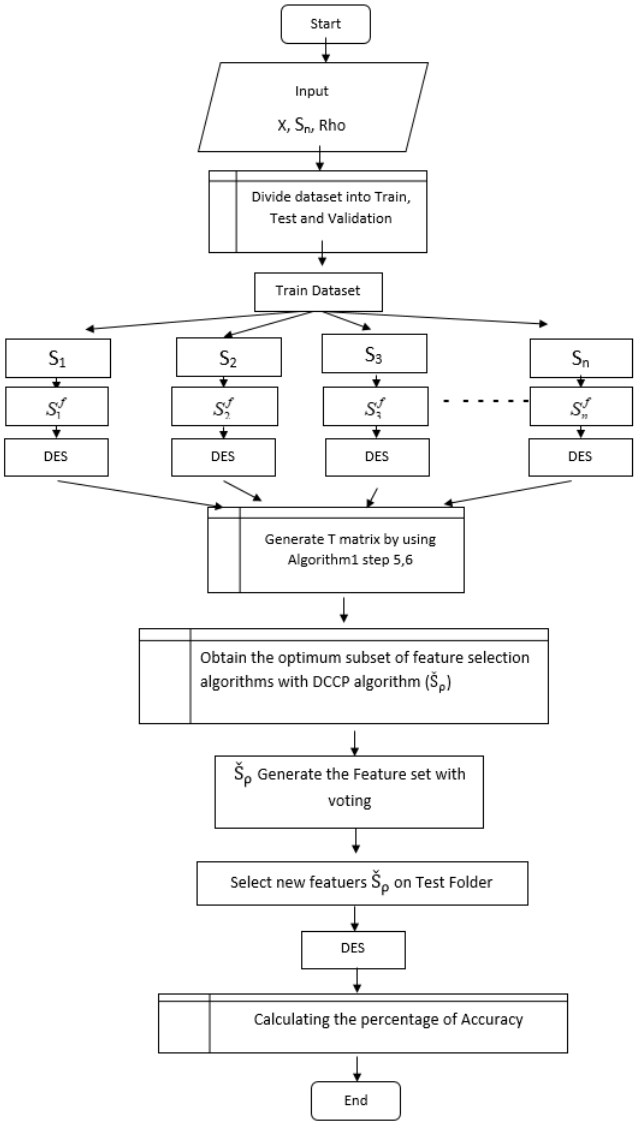

Figure 1: Flow chart of the proposed method

The percentage of accuracy results on the test set for the proposed model PrunedOPT is given in the first column of Table 2 and accuracy values of the unpruned case corresponding to Full Ensemble, and the Joint Criterion with its best pruning rates are illustrated by second and the third columns respectively. Here, the best pruning rate of Joint Criterion is selected among the values $[5, 10, 15, 20]$ based on corresponding accuracies. It is clear from the experimental results that the proposed ensemble based feature selection approach PrunedOPT achieves better prediction accuracies than both unpruned case and Joint Criterion. In Table 2, the bold numbers correspond to the best accuracy values measured by the ratio of correct predictions to the total number of examples in the test set.

In order to perform a thorough comparison of the results, we presented these results in Tables 2 and visually by Figure 2 for each of the eight data sets. In Figure 2, each subfigure stands for different data sets in which our proposed method called PrunedOPT and the method Joint Criterion are compared against their accuracy values versus pruning rate k. It is clear from each subfigure in Figure 2 that the proposed optimization model PrunedOPT approximates the optimal accuracy value with respect to its optimal pruning rate when compared with accuracy values corresponding to various pruning rates of Joint Criterion method.

Dataset	Ensemble Feature Selection PrunedOPT by DCCP	Ensemble Feature Selection Unpruned Case	Joint Criterion (Pruning Rate)
Lung small	**0.740**	0.712	0.711 (5)
Madelon	**0.653**	0.580	0.578 (20)
Yale	**0.655**	0.467	0.422(20)
WarpAR10P	**0.741**	0.642	0.683 (20)
Colon	**0.759**	0.690	0.664 (20)
Urban Land Cover	**0.680**	0.591	0.40 (5)
Libras Movement	**0.701**	0.658	0.50 (20)
Hill-Valley	**0.690**	0.667	0.632 (20)

Table 2: Proposed Feature Selection Algorithm percentage of accuracy.

As the Joint Criterion method requires pruning rate to be an input parameter for each different input parameters, we get different accuracy values. In order to compare the best accuracy results of the Joint Criterion method with the proposed ensemble pruning, we illustrated the best of each method for each data referring to their corresponding pruning rates in Table 2. It should be noted that our proposed approach achieves better accuracy values than both unpruned case and Joint Criterion.

For each of the 8 datasets, the performance evaluation was measured against each of those 28 constituent feature selection methods. Those methods were run 5 times randomly. The performance (accuracy) results, which are calculated by DES, of these methods are shown in Table 4 -11 where the first column represents the method names, remaning second, third, fourth and fifth columns show the accuracy values calculated by using equation (9) below:

$$Accuracy = \frac{\#\ of\ correct\ predictions}{\#\ of\ samples}. \tag{9}$$

The last two columns named by AVG and STD show the average accuracy and average standard deviation values of 5 random iterations for each data set.

The average running time required for DCCP were calculated as two minutes while time required for the Joint Criterion method was three seconds. From these results, we observe that our proposed ensemble pruning method takes longer time than Joint Criterion. However, this drawback is not a fair comparison between two methods since the proposed pruning method gives not only higher accuracy but it also automates finding the optimal pruning rate with a unified framework of optimizing accuracy and diversity simultaneously; whereas the joint criterion finds the pruning rate combinatorially. Thus, when one compares the running times, time spend for trials of different selections of pruning rates should be considered. Further, one cannot know the optimal pruning rate without trials in Joint Criterion which differs also in each data set.

In our experimental analysis, the selected features by PrunedOPT, unpruned case and Joint Criterion for each data set were tested with widely used ML methods such as decision tree algorithm, nonlinear SVM and DES in Table

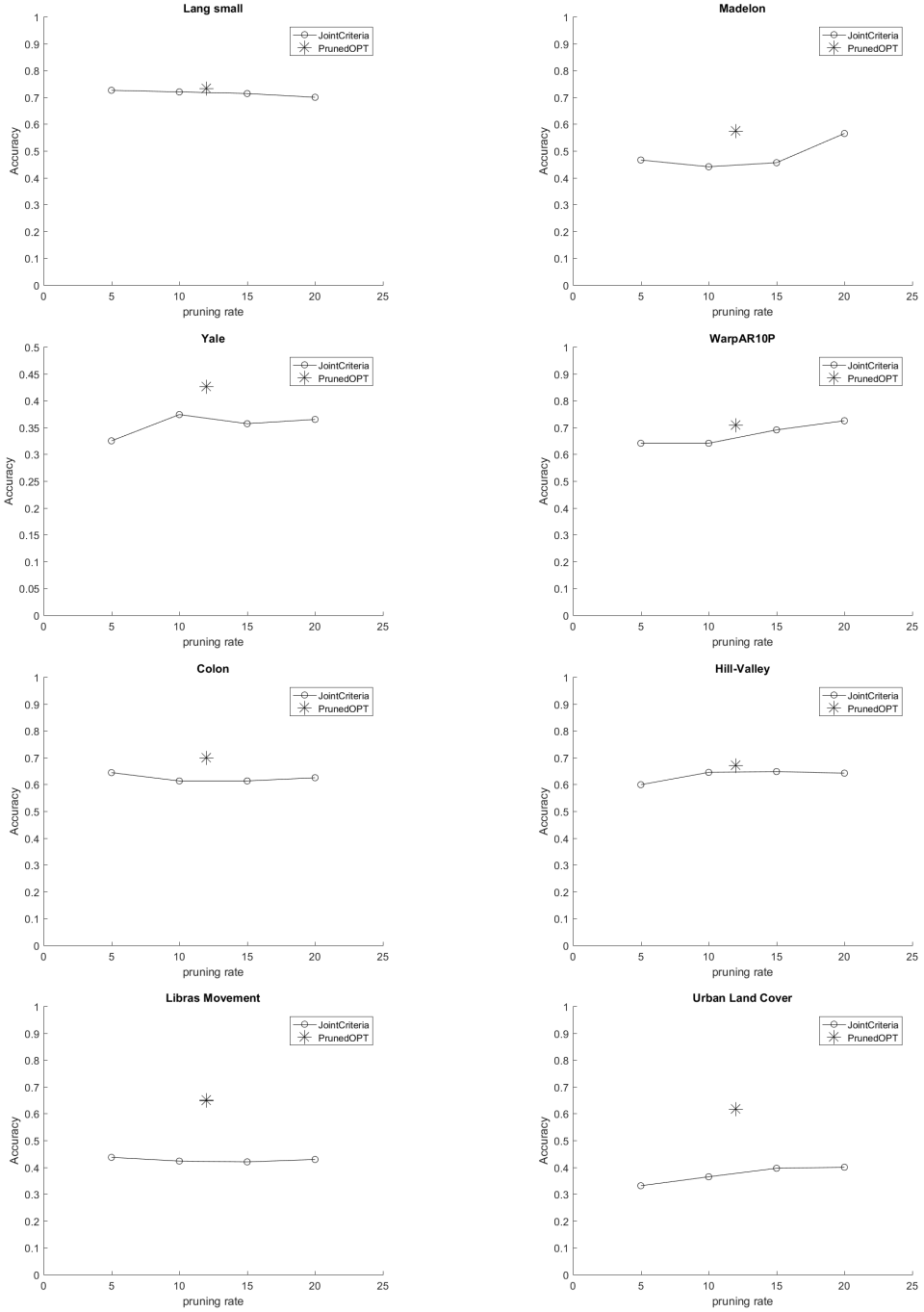

Figure 2: Graphical illustration of NMI values versus various pruning rates for all data sets.

Methods	PrunedOPT by DCCP	Unpruned Case	Joint Criterion
Linear SVM	**0.634**	0.615	0.557
Non-linear SVM	**0.625**	0.612	0.608
Decision Tree	0.608	**0.610**	0.578
DES	**0.702**	0.625	0.557

Table 3: Proposed Feature Selection Algorithm percentage of accuracy.

3. It is clear that best performance was obtained by DES which is the main reason that why we selected it as a classification method in this study. It should be noted that the proposed algorithm PrunedOPT has always better accuracy values in average for all data sets in Table 3 when compared with other methods for all classification algorithms except decision tree with a slight difference. The reason behind this can come from the well-known fact that there is no unique and best algorithm that works for all type of data sets. This slight difference can be seen as negligible.

Data	**Accuracy**						
Small Lung Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,53	0,21	0,33	0,00	0,32	0,28	0,19
(SPEC)	0,63	0,47	0,58	0,57	0,32	0,52	0,13
Fisher Score	0,42	0,37	0,67	0,29	0,47	0,44	0,14
Trace Ratio Criterion	0,32	0,32	0,00	0,86	0,21	0,34	0,32
ReliefF	0,16	0,53	0,58	0,43	0,68	0,48	0,20
MIM	0,58	0,42	0,42	0,29	0,47	0,44	0,11
MIFS	0,53	0,11	0,25	0,14	0,37	0,28	0,17
MRMR	0,47	0,32	0,50	0,57	0,68	0,51	0,14
CIFE	0,32	0,26	0,33	0,57	0,68	0,43	0,18
JMI	0,53	0,32	0,42	0,57	0,21	0,41	0,15
CMIM	0,21	0,42	0,17	0,43	0,32	0,31	0,12
DISR	0,53	0,26	0,17	0,29	0,47	0,34	0,15
FCBF	0,32	0,21	0,50	0,71	0,53	0,45	0,20
Interaction Capping	0,63	0,37	0,33	0,57	0,11	0,40	0,21
MCFS	0,26	0,32	0,33	0,86	0,37	0,43	0,24
L1 norm Regularization	0,47	0,26	0,58	0,43	0,47	0,44	0,12
l2;1 norm Regularized	0,37	0,11	0,42	0,57	0,47	0,39	0,17
NDFS	0,26	0,63	0,67	0,71	0,37	**0,53**	0,20
F-score	0,32	0,26	0,17	0,29	0,32	0,27	0,06
Gini Index	0,37	0,21	0,42	0,57	0,42	0,40	0,13
CFS	0,42	0,21	0,58	0,57	0,42	0,44	0,15
Wraper	0,53	0,21	0,25	0,14	0,26	0,28	0,15
Group Feature Structures	0,47	0,26	0,50	0,43	0,21	0,38	0,13
UDFS	0,26	0,32	0,67	0,29	0,21	0,35	0,18
Tree-fs	0,42	0,21	0,50	0,57	0,42	0,42	0,14
RFS	0,16	0,63	0,33	0,57	0,26	0,39	0,20
SVMBackward	0,63	0,26	0,50	0,71	0,26	0,47	0,21
SVMForward	0,32	0,42	0,50	0,57	0,32	0,42	0,11
AVG	0,40	0,35	0,41	0,48	0,37	0,40	0,16

Table 4: The accuracy measurements for each feature selection algorithms for Small lung dataset.

Data Set	**Accuracy**						
Madelon Data	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,38	0,60	0,50	0,56	0,30	0,47	0,13
(SPEC)	0,38	0,50	0,17	0,56	0,50	0,42	0,16
Fisher Score	0,38	0,30	0,33	0,75	0,10	0,37	0,24
Trace Ratio Criterion	0,38	0,50	0,67	0,44	0,20	0,44	0,17
ReliefF	0,31	0,50	0,50	0,69	0,40	0,48	0,14
MIM	0,75	0,30	0,33	0,63	0,70	0,54	0,21
MIFS	0,63	0,60	0,17	0,56	0,60	0,51	0,19
MRMR	0,63	0,50	0,33	0,50	0,40	0,47	0,11
CIFE	0,38	0,40	0,50	0,50	0,40	0,44	0,06
JMI	0,56	0,60	0,33	0,56	0,70	0,55	0,13
CMIM	0,44	0,50	0,33	0,63	0,50	0,48	0,11
DISR	0,56	0,40	0,67	0,81	0,60	0,61	0,15
FCBF	0,56	0,40	0,50	0,69	0,40	0,51	0,12
Interaction Capping	0,44	0,60	0,33	0,56	0,70	0,53	0,14
MCFS	0,69	0,50	0,33	0,63	0,50	0,53	0,14
L1 norm Regularization	0,44	0,70	0,33	0,69	0,40	0,51	0,17
l2;1 norm Regularized	0,50	0,40	0,50	0,56	0,50	0,49	0,06
NDFS	0,44	0,60	0,83	0,88	0,60	**0,67**	0,18
F-score	0,56	0,20	0,33	0,63	0,40	0,42	0,17
Gini Index	0,63	0,40	0,33	0,69	0,60	0,53	0,15
CFS	0,50	0,20	0,33	0,69	0,20	0,38	0,21
Wraper	0,31	0,50	0,17	0,56	0,30	0,37	0,16
Group Feature Structures	0,44	0,30	0,50	0,81	0,30	0,47	0,21
UDFS	0,44	0,50	0,33	0,69	0,40	0,47	0,13
Tree-fs	0,44	0,10	0,83	0,50	0,60	0,49	0,27
RFS	0,44	0,30	0,17	0,56	0,40	0,37	0,15
SVMBackward	0,56	0,50	0,33	0,63	0,50	0,50	0,11
SVMForward	0,50	0,50	0,33	0,50	0,10	0,39	0,18
AVG	0,48	0,47	0,40	0,62	0,43	0,47	0,15

Table 5: The accuracy measurements for each feature selection algorithms for Madelon dataset.

Data	**Accuracy**						
Yale Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,31	0,30	0,12	0,55	0,86	0,43	0,29
(SPEC)	0,17	0,37	0,41	0,64	0,57	0,43	0,18
Fisher Score	0,36	0,44	0,53	0,36	0,57	0,45	0,10
Trace Ratio Criterion	0,12	0,26	0,41	0,45	0,29	0,31	0,13
ReliefF	0,24	0,33	0,24	0,36	0,29	0,29	0,06
MIM	0,19	0,44	0,47	0,55	0,71	0,47	0,19
MIFS	0,24	0,37	0,29	0,36	0,43	0,34	0,07
MRMR	0,36	0,22	0,65	0,64	0,57	0,49	0,19
CIFE	0,33	0,30	0,47	0,82	0,43	0,47	0,21
JMI	0,40	0,48	0,59	0,82	0,57	**0,57**	0,16
CMIM	0,24	0,48	0,47	0,36	0,43	0,40	0,10
DISR	0,29	0,44	0,59	0,64	0,71	0,53	0,17
FCBF	0,33	0,41	0,47	0,64	0,57	0,48	0,12
Interaction Capping	0,29	0,48	0,59	0,36	1,00	0,54	0,28
MCFS	0,19	0,33	0,12	0,27	0,57	0,30	0,17
L1 norm Regularization	0,26	0,41	0,65	0,55	0,57	0,49	0,15
l2;1 norm Regularized	0,38	0,37	0,41	0,82	0,71	0,54	0,21
NDFS	0,33	0,33	0,24	0,55	0,57	0,40	0,15
F-score	0,24	0,22	0,24	0,27	0,29	0,25	0,03
Gini Index	0,29	0,37	0,41	0,36	0,43	0,37	0,06
CFS	0,17	0,19	0,35	0,18	0,57	0,29	0,17
Wraper	0,31	0,52	0,53	0,45	0,29	0,42	0,12
Group Feature Structures	0,14	0,48	0,53	0,27	0,86	0,46	0,27
UDFS	0,17	0,56	0,59	0,45	0,57	0,47	0,18
Tree-fs	0,26	0,41	0,53	0,45	0,14	0,36	0,16
RFS	0,33	0,48	0,24	0,27	0,71	0,41	0,20
SVMBackward	0,24	0,30	0,47	0,64	0,86	0,50	0,25
SVMForward	0,31	0,41	0,35	0,18	0,57	0,36	0,14
AVG	0,26	0,42	0,42	0,47	0,56	0,42	0,16

Table 6: The accuracy measurements for each feature selection algorithms for Yale dataset.

Data	**Accuracy**						
Warp Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,38	0,31	0,30	0,13	0,13	0,25	0,12
(SPEC)	0,44	0,25	0,70	0,50	0,31	0,44	0,18
Fisher Score	0,50	0,44	0,90	0,56	0,63	0,61	0,18
Trace Ratio Criterion	0,44	0,56	0,90	0,19	0,69	0,56	0,27
ReliefF	0,50	0,44	0,60	0,25	0,19	0,40	0,17
MIM	0,69	0,69	0,80	0,50	0,63	0,66	0,11
MIFS	0,13	0,31	0,60	0,50	0,31	0,37	0,18
MRMR	0,56	0,38	0,80	0,56	0,44	0,55	0,16
CIFE	0,56	0,25	0,60	0,56	0,19	0,43	0,20
JMI	0,75	0,50	0,70	0,56	0,50	0,60	0,12
CMIM	0,56	0,56	0,50	0,38	0,50	0,50	0,08
DISR	0,69	0,75	0,60	0,31	0,44	0,56	0,18
FCBF	0,19	0,63	1,00	0,75	0,75	0,66	0,30
Interaction Capping	0,56	0,56	0,70	0,38	0,50	0,54	0,12
MCFS	0,69	0,56	0,90	0,69	0,56	0,68	0,14
L1 norm Regularization	0,50	0,19	0,70	0,56	0,19	0,43	0,23
l2;1 norm Regularized	0,88	0,50	0,50	0,69	0,56	**0,63**	0,16
NDFS	0,88	0,38	0,80	0,50	0,56	0,62	0,21
F-score	0,44	0,25	0,60	0,50	0,50	0,46	0,13
Gini Index	0,06	0,44	0,60	0,44	0,25	0,36	0,21
CFS	0,50	0,88	0,90	0,19	0,31	0,56	0,32
Wraper	0,38	0,50	0,50	0,50	0,31	0,44	0,09
Group Feature Structures	0,56	0,38	0,90	0,56	0,19	0,52	0,26
UDFS	0,63	0,06	0,70	0,50	0,44	0,47	0,25
Tree-fs	0,38	0,88	0,70	0,38	0,19	0,50	0,28
RFS	0,44	0,69	0,70	0,38	0,38	0,52	0,17
SVMBackward	0,63	0,31	0,90	0,63	0,31	0,56	0,25
SVMForward	0,38	0,63	0,50	0,88	0,19	0,51	0,26
AVG	0,51	0,51	0,7	0,48	0,39	0,51	0,19

Table 7: The accuracy measurements for each feature selection algorithms for Warp dataset.

Data	**Accuracy**						
Colon Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,31	0,88	0,80	0,50	0,44	0,59	0,24
(SPEC)	0,56	0,75	0,80	0,44	0,88	0,69	0,18
Fisher Score	0,81	1,00	0,90	0,63	0,94	0,86	0,15
Trace Ratio Criterion	0,88	0,94	1,00	0,75	0,69	0,85	0,13
ReliefF	0,88	1,00	0,50	0,69	0,81	0,78	0,19
MIM	0,88	1,00	0,90	0,75	0,88	**0,88**	0,09
MIFS	0,63	0,94	0,50	0,44	0,63	0,63	0,19
MRMR	0,56	0,94	0,80	0,63	0,88	0,76	0,16
CIFE	1,00	0,88	0,40	0,88	0,88	0,81	0,23
JMI	0,94	1,00	0,80	0,63	0,88	0,85	0,14
CMIM	0,69	0,88	1,00	0,69	0,81	0,81	0,13
DISR	0,94	0,63	1,00	0,88	0,88	0,86	0,14
FCBF	0,88	0,94	1,00	0,88	0,88	0,91	0,06
Interaction Capping	0,56	0,69	0,40	0,31	0,38	0,47	0,15
MCFS	0,88	0,75	0,50	0,44	0,88	0,69	0,21
L1 norm Regularization	0,50	0,50	0,50	0,75	0,56	0,56	0,11
l2;1 norm Regularized	0,94	0,75	0,90	0,56	0,88	0,81	0,15
NDFS	0,88	0,81	0,60	0,56	0,75	0,72	0,13
F-score	0,63	0,44	0,50	0,56	0,75	0,58	0,12
Gini Index	0,81	0,69	0,60	0,56	0,56	0,65	0,11
CFS	0,50	0,88	0,50	0,88	0,81	0,71	0,20
Wraper	0,56	0,75	0,90	0,81	0,38	0,68	0,21
Group Feature Structures	0,75	0,88	0,80	0,56	0,69	0,74	0,12
UDFS	0,56	0,81	0,70	0,63	0,81	0,70	0,11
Tree-fs	0,94	0,69	0,80	0,56	0,69	0,74	0,14
RFS	0,81	0,50	0,80	0,50	0,81	0,69	0,17
SVMBackward	0,69	0,63	0,50	0,81	0,69	0,66	0,11
SVMForward	0,88	0,69	0,60	0,63	0,81	0,72	0,12
AVG	0,74	0,83	0,71	0,63	0,74	0,72	0,14

Table 8: The accuracy measurements for each feature selection algorithms for Colon dataset.

Data	**Accuracy**						
Urban Land Cover Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,56	0,64	0,61	0,49	0,70	0,60	0,08
(SPEC)	0,47	0,70	0,70	0,61	0,35	0,57	0,15
Fisher Score	0,88	0,64	0,65	0,44	0,85	0,69	0,18
Trace Ratio Criterion	0,70	0,58	0,61	0,61	0,55	0,61	0,06
ReliefF	0,65	0,51	0,65	0,38	0,45	0,53	0,12
MIM	0,65	0,45	0,38	0,44	0,60	0,50	0,12
MIFS	0,52	0,58	0,70	0,32	0,45	0,51	0,14
MRMR	0,79	0,58	0,43	0,55	0,50	0,57	0,14
CIFE	0,65	0,58	0,79	0,44	0,60	0,61	0,13
JMI	0,70	0,58	0,56	0,61	0,35	0,56	0,13
CMIM	0,61	0,51	0,70	0,61	0,50	0,59	0,08
DISR	0,65	0,95	0,75	0,49	0,35	0,64	0,23
FCBF	0,65	0,64	0,84	0,38	0,75	0,65	0,17
Interaction Capping	0,79	0,70	0,70	0,73	0,65	**0,71**	0,05
MCFS	0,56	0,58	0,52	0,26	0,70	0,52	0,16
L1 norm Regularization	0,56	0,58	0,75	0,55	0,65	0,62	0,08
l2;1 norm Regularized	0,56	0,64	0,75	0,49	0,55	0,60	0,10
NDFS	0,79	0,58	0,65	0,49	0,45	0,59	0,14
F-score	0,65	0,58	0,56	0,49	0,55	0,57	0,06
Gini Index	0,52	0,51	0,70	0,44	0,50	0,53	0,10
CFS	0,56	0,58	0,61	0,38	0,50	0,52	0,09
Wraper	0,47	0,51	0,65	0,61	0,55	0,56	0,07
Group Feature Structures	0,52	0,45	0,56	0,32	0,50	0,47	0,09
UDFS	0,61	0,70	0,70	0,61	0,40	0,60	0,12
Tree-fs	0,70	0,64	0,70	0,67	0,45	0,63	0,10
RFS	0,79	0,51	0,52	0,44	0,60	0,57	0,14
SVMBackward	0,70	0,76	0,43	0,61	0,70	0,64	0,13
SVMForward	0,61	0,45	0,56	0,55	0,50	0,54	0,06
AVG	0.63	0,59	0,63	0,50	0,54	0,58	0,11

Table 9: The accuracy measurements for each feature selection algorithms for Urban Land Cover dataset.

Data	**Accuracy**						
Libras Movement Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,49	0,30	0,72	0,67	0,56	0,55	0,17
(SPEC)	0,36	0,36	0,88	0,76	0,56	0,58	0,23
Fisher Score	0,43	0,59	0,62	0,62	0,69	0,59	0,10
Trace Ratio Criterion	0,36	0,36	0,62	0,76	0,69	0,56	0,19
ReliefF	0,61	0,30	0,35	0,29	0,50	0,41	0,14
MIM	0,30	0,54	0,56	0,71	0,56	0,54	0,15
MIFS	0,43	0,42	0,56	0,67	0,69	0,55	0,13
MRMR	0,43	0,36	0,72	0,90	0,75	0,63	0,23
CIFE	0,49	0,36	0,46	0,67	0,75	0,54	0,16
JMI	0,43	0,48	0,41	0,67	0,69	0,53	0,14
CMIM	0,43	0,36	0,51	0,52	0,63	0,49	0,10
DISR	0,61	0,48	0,83	0,52	0,75	**0,64**	0,15
FCBF	0,43	0,30	0,77	0,43	0,50	0,49	0,18
Interaction Capping	0,49	0,42	0,51	0,52	0,50	0,49	0,04
MCFS	0,49	0,42	0,46	0,62	0,69	0,53	0,11
L1 norm Regularization	0,43	0,36	0,72	0,57	0,56	0,53	0,14
l2;1 norm Regularized	0,49	0,48	0,72	0,62	0,69	0,60	0,11
NDFS	0,74	0,36	0,51	0,76	0,75	0,62	0,18
F-score	0,61	0,30	0,62	0,67	0,63	0,56	0,15
Gini Index	0,30	0,48	0,67	0,62	0,50	0,51	0,14
CFS	0,55	0,42	0,51	0,57	0,50	0,51	0,06
Wraper	0,43	0,48	0,67	0,71	0,69	0,59	0,13
Group Feature Structures	0,49	0,36	0,62	0,62	0,56	0,53	0,11
UDFS	0,49	0,48	0,51	0,71	0,56	0,55	0,10
Tree-fs	0,49	0,30	0,62	0,57	0,63	0,52	0,13
RFS	0,61	0,48	0,67	0,48	0,63	0,57	0,09
SVMBackward	0,49	0,42	0,77	0,71	0,81	0,64	0,18
SVMForward	0,43	0,42	0,67	0,43	0,56	0,50	0,11
AVG	0,47	0,42	0,61	0,62	0,62	0,54	0.13

Table 10: The accuracy measurements for each feature selection algorithms for Libras Movement dataset.

Data	**Accuracy**						
Hill-Valley Dataset	1st	2nd	3rd	4th	5th	AVG	STD
LS	0,63	0,33	0,50	0,31	0,50	0,45	0,13
(SPEC)	0,56	0,27	0,44	0,44	0,63	0,47	0,14
Fisher Score	0,50	0,53	0,44	0,44	0,50	0,48	0,04
Trace Ratio Criterion	0,63	0,40	0,56	0,31	0,44	0,47	0,13
ReliefF	0,63	0,27	0,56	0,56	0,50	0,50	0,14
MIM	0,50	0,60	0,63	0,50	0,69	**0,58**	0,08
MIFS	0,44	0,53	0,38	0,56	0,38	0,46	0,09
MRMR	0,50	0,40	0,44	0,50	0,56	0,48	0,06
CIFE	0,69	0,53	0,44	0,50	0,63	0,56	0,10
JMI	0,50	0,40	0,63	0,63	0,44	0,52	0,10
CMIM	0,44	0,40	0,38	0,50	0,44	0,43	0,05
DISR	0,63	0,60	0,38	0,50	0,69	0,56	0,12
FCBF	0,44	0,53	0,56	0,25	0,50	0,46	0,12
Interaction Capping	0,88	0,67	0,38	0,44	0,44	0,56	0,21
MCFS	0,81	0,60	0,38	0,38	0,63	0,56	0,19
L1 norm Regularization	0,69	0,60	0,44	0,31	0,56	0,52	0,15
l2;1 norm Regularized	0,63	0,40	0,63	0,63	0,56	0,57	0,10
NDFS	0,63	0,53	0,44	0,31	0,50	0,48	0,12
F-score	0,63	0,40	0,38	0,63	0,31	0,47	0,15
Gini Index	0,50	0,53	0,38	0,44	0,69	0,51	0,12
CFS	0,56	0,40	0,44	0,56	0,56	0,51	0,08
Wraper	0,50	0,33	0,56	0,56	0,31	0,45	0,12
Group Feature Structures	0,75	0,40	0,31	0,50	0,56	0,51	0,17
UDFS	0,56	0,40	0,63	0,38	0,63	0,52	0,12
Tree-fs	0,75	0,60	0,44	0,50	0,19	0,50	0,21
RFS	0,56	0,40	0,44	0,38	0,38	0,43	0,08
SVMBackward	0,63	0,33	0,56	0,50	0,69	0,54	0,14
SVMForward	0,63	0,27	0,56	0,56	0,50	0,50	0,14
AVG	0,6	0.50	0,47	0,46	0,51	0.50	0.12

Table 11: The accuracy measurements for each feature selection algorithms for Hill-Valley dataset.

6. Discussion and Conclusion

In this paper, a novel ensemble learning based feature selection is proposed which automatically calculates the best subset of feature selectors considering the accuracy and diversity trade off with an optimization framework by DCCP algorithm. The proposed approach is validated on the most well known data sets and the performance results are compared with an un-pruned case of ensemble learning and Joint criterion method. DES is used as a classifier of these classification tasks. In addition to this, we implmented our model with other classification algorithms such as Linear SVM, nonlinear SVM and decision tree method where our proposed approach gave better accuracy performance with those classification techniques as well. As a future study, ensemble library can be enhanced by considering data variation techniques such as bagging.

The performance evaluation was carried out against each of those 28 constituent feature selection methods. When analyzing feature selection algorithms individually versus Ensemble Feature selection method performance, ensemble methods show great promise for large feature domains. It turns out that the best trade-off between accuracy and diversity performance depends on the ensemble feature selection model, giving rise to a new model selection strategy.

7. Credit Author Statement

Pınar Karadayı Ataş: Software, Validation, Methodology, Writing- Original draft preparation. **Süreyya Özöğür - Akyüz:** Conceptualization, Methodology, Reviewing and Editing.

References

[1] Iñaki Inza, Pedro Larrañaga, Rosa Blanco, and Antonio J Cerrolaza. Filter versus wrapper gene selection approaches in dna microarray domains. Artificial intelligence in medicine, 31(2):91–103, 2004.

[2] George Forman. An extensive empirical study of feature selection metrics for text classification. Journal of machine learning research, 3(Mar):1289–1305, 2003.

[3] Verónica Bolón-Canedo, Noelia Sánchez-Maroño, and Amparo Alonso-Betanzos. Feature selection for high-dimensional data. Progress in Artificial Intelligence, 5(2):65–75, 2016.

[4] Süreyya Özöğür-Akyüz, Terry Windeatt, and Raymond Smith. Pruning of error correcting output codes by optimization of accuracy–diversity trade off. Machine Learning, 101(1-3):253–269, 2015.

[5] Yi Zhang, Samuel Burer, and W Nick Street. Ensemble pruning via semi-definite programming. Journal of Machine Learning Research, 7(Jul):1315–1338, 2006.

[6] Leo Breiman. Bagging predictors. Machine learning, 24(2):123–140, 1996.

[7] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1):119–139, 1997.

[8] Iñigo Barandiaran. The random subspace method for constructing decision forests. IEEE transactions on pattern analysis and machine intelligence, 20(8), 1998.

[9] Donghai Guan, Weiwei Yuan, Young-Koo Lee, Kamran Najeebullah, and Mostofa Kamal Rasel. A review of ensemble learning based feature selection. IETE Technical Review, 31(3):190–198, 2014.

[10] David J Dittman, Taghi M Khoshgoftaar, Randall Wald, and Amri Napolitano. Comparing two new gene selection ensemble approaches with the commonly-used approach. In Machine Learning and Applications (ICMLA), 2012 11th International Conference on, volume 2, pages 184–191. IEEE, 2012.

[11] Padraig Cunningham and John Carney. Diversity versus quality in classification ensembles based on feature selection. In European Conference on Machine Learning, pages 109–116. Springer, 2000.

[12] Yvan Saeys, Thomas Abeel, and Yves Van de Peer. Robust feature selection using ensemble feature selection techniques. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 313–325. Springer, 2008.

[13] Borja Seijo-Pardo, Iago Porto-Díaz, Verónica Bolón-Canedo, and Amparo Alonso-Betanzos. Ensemble feature selection: homogeneous and heterogeneous approaches. Knowledge-Based Systems, 118:124–139, 2017.

[14] Asit K Das, Sunanda Das, and Arka Ghosh. Ensemble feature selection using bi-objective genetic algorithm. Knowledge-Based Systems, 123:116–127, 2017.

[15] Borja Seijo-Pardo, Verónica Bolón-Canedo, and Amparo Alonso-Betanzos. Testing different ensemble configurations for feature selection. Neural Processing Letters, 46(3):857–880, 2017.

[16] Alexey Tsymbal, Seppo Puuronen, and David W Patterson. Ensemble feature selection with the simple bayesian classification. Information fusion, 4(2):87–100, 2003.

[17] Verónica Bolón-Canedo, Noelia Sánchez-Maroño, and Amparo Alonso-Betanzos. An ensemble of filters and classifiers for microarray data classification. Pattern Recognition, 45(1):531–539, 2012.

[18] Verónica Bolón-Canedo, Noelia Sánchez-Marono, and Amparo Alonso-Betanzos. Data classification using an ensemble of filters. Neurocomputing, 135:13–20, 2014.

[19] Feng Yang and KZ Mao. Robust feature selection for microarray data based on multicriterion fusion. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 8(4):1080–1092, 2010.

[20] Terry Windeatt, Rakkrit Duangsoithong, and Raymond Smith. Embedded feature ranking for ensemble mlp classifiers. IEEE transactions on neural networks, 22(6):988–994, 2011.

[21] Huanjing Wang, Taghi M Khoshgoftaar, and Amri Napolitano. A comparative study of ensemble feature selection techniques for software defect prediction. In 2010 Ninth International Conference on Machine Learning and Applications, pages 135–140. IEEE, 2010.

[22] Thomas Abeel, Thibault Helleputte, Yves Van de Peer, Pierre Dupont, and Yvan Saeys. Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. Bioinformatics, 26(3):392–398, 2009.

[23] Afef Ben Brahim and Mohamed Limam. Robust ensemble feature selection for high dimensional data sets. In 2013 International Conference on High Performance Computing & Simulation (HPCS), pages 151–157. IEEE, 2013.

[24] Borja Seijo-Pardo, Verónica Bolón-Canedo, and Amparo Alonso-Betanzos. On developing an automatic threshold applied to feature selection ensembles. Information Fusion, 45:227–245, 2019.

[25] Lawrence Mitchell, Terence M Sloan, Muriel Mewissen, Peter Ghazal, Thorsten Forster, Michal Piotrowski, and Arthur Trew. Parallel classification and feature selection in microarray data using sprint. Concurrency and computation: practice and experience, 26(4):854–865, 2014.

[26] Carlos Eiras-Franco, Verónica Bolón-Canedo, Sabela Ramos, Jorge González-Domínguez, Amparo Alonso-Betanzos, and Juan Tourino. Multithreaded and spark parallelization of feature selection filters. Journal of Computational Science, 17:609–619, 2016.

[27] Yi Hong, Sam Kwong, Yuchou Chang, and Qingsheng Ren. Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm. Pattern Recognition, 41(9):2742–2756, 2008.

[28] Yi Hong, Sam Kwong, Yuchou Chang, and Qingsheng Ren. Consensus unsupervised feature ranking from multiple views. Pattern Recognition Letters, 29(5):595–602, 2008.

[29] Marisa Morita, Luiz S Oliveira, and Robert Sabourin. Unsupervised feature selection for ensemble of classifiers. In Ninth International Workshop on Frontiers in Handwriting Recognition, pages 81–86. IEEE, 2004.

[30] Fazia Bellal, Haytham Elghazel, and Alex Aussem. A semi-supervised feature ranking method with ensemble learning. Pattern Recognition Letters, 33(10):1426–1433, 2012.

[31] Yongkoo Han, Kisung Park, and Young-Koo Lee. Confident wrapper-type semi-supervised feature selection using an ensemble classifier. In 2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), pages 4581–4586. IEEE, 2011.

[32] Albert HR Ko, Robert Sabourin, and Alceu Souza Britto Jr. From dynamic classifier selection to dynamic ensemble selection. Pattern recognition, 41(5):1718–1731, 2008.

[33] Michael Grant, Stephen Boyd, and Yinyu Ye. Disciplined convex programming. In Global optimization, pages 155–210. Springer, 2006.

[34] Xinyue Shen, Steven Diamond, Yuantao Gu, and Stephen Boyd. Disciplined convex-concave programming. In Decision and Control (CDC), 2016 IEEE 55th Conference on, pages 1009–1014. IEEE, 2016.

[35] Ludmila I Kuncheva. A theoretical study on six classifier fusion strategies. IEEE Transactions on pattern analysis and machine intelligence, 24(2):281–286, 2002.

[36] Rodrigo GF Soares, Alixandre Santana, Anne MP Canuto, and Marcílio Carlos Pereira de Souto. Using accuracy and diversity to select classifiers to build ensembles. In The 2006 IEEE International Joint Conference on Neural Network Proceedings, pages 1310–1316. IEEE, 2006.

[37] Yue Zhao, Xuejian Wang, Cheng Cheng, and Xueying Ding. Combining machine learning models using combo library. arXiv preprint arXiv:1910.07988, 2019.

[38] Xiaoli Z. Fern and Wei Lin. Cluster ensemble selection. Stat. Anal. Data Min., 1(3):128–141, November 2008.

[39] Xiaofei He, Deng Cai, and Partha Niyogi. Laplacian score for feature selection. In Advances in neural information processing systems, pages 507–514, 2006.

[40] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Advances in neural information processing systems, pages 585–591, 2002.

[41] X He and P Niyogi. Locality preserving projection, neural information processing symposium, vancouver. British Columbia, Canada, 2003.

[42] Zheng Zhao and Huan Liu. Spectral feature selection for supervised and unsupervised learning. In Proceedings of the 24th international conference on Machine learning, pages 1151–1157. ACM, 2007.

[43] Richard O Duda, Peter E Hart, and David G Stork. Pattern classification. John Wiley & Sons, 2012.

[44] Feiping Nie, Shiming Xiang, Yangqing Jia, Changshui Zhang, and Shuicheng Yan. Trace ratio criterion for feature selection. In AAAI, volume 2, pages 671–676, 2008.

[45] Marko Robnik-Šikonja and Igor Kononenko. Theoretical and empirical analysis of relieff and rrelieff. Machine learning, 53(1-2):23–69, 2003.

[46] David D Lewis. Feature selection and feature extraction for text categorization. In Proceedings of the workshop on Speech and Natural Language, pages 212–217. Association for Computational Linguistics, 1992.

[47] Roberto Battiti. Using mutual information for selecting features in supervised neural net learning. IEEE Transactions on neural networks, 5(4):537–550, 1994.

[48] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on pattern analysis and machine intelligence, 27(8):1226–1238, 2005.

[49] Lin Tang, Zhu Lin, and Yong-ming Li. Effects of different magnitudes of mechanical strain on osteoblasts in vitro. Biochemical and biophysical research communications, 344(1):122–128, 2006.

[50] Howard Hua Yang and John Moody. Data visualization and feature selection: New algorithms for nongaussian data. In Advances in Neural Information Processing Systems, pages 687–693, 2000.

[51] Michel Vidal-Naquet and Shimon Ullman. Object recognition with informative features and linear classification. In ICCV, volume 3, page 281, 2003.

[52] François Fleuret. Fast binary feature selection with conditional mutual information. Journal of Machine Learning Research, 5(Nov):1531–1555, 2004.

[53] Patrick E Meyer and Gianluca Bontempi. On the use of variable complementarity for feature selection in cancer classification. In Workshops on applications of evolutionary computation, pages 91–102. Springer, 2006.

[54] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In Proceedings of the 20th international conference on machine learning (ICML-03), pages 856–863, 2003.

[55] Aleks Jakulin. Machine learning based on attribute interactions: phd dissertation. PhD thesis, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, 2005.

[56] J. Ross Quinlan. Induction of decision trees. Machine learning, 1(1):81–106, 1986.

[57] J Ross Quinlan. C4. 5: programs for machine learning. Mach. Learn, 16(3):235–240, 1993.

[58] Liang Du and Yi-Dong Shen. Unsupervised feature selection with adaptive structure learning. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, pages 209–218. ACM, 2015.

[59] Xinwang Liu, Lei Wang, Jian Zhang, Jianping Yin, and Huan Liu. Global and local structure preservation for feature selection. IEEE Transactions on Neural Networks and Learning Systems, 25(6):1083–1095, 2014.

[60] Deng Cai, Chiyuan Zhang, and Xiaofei He. Unsupervised feature selection for multi-cluster data. In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 333–342. ACM, 2010.

[61] Robert Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological), pages 267–288, 1996.

[62] Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, Hanqing Lu, et al. Unsupervised feature selection using nonnegative spectral analysis. In AAAI, volume 2, pages 1026–1032, 2012.

[63] Sewall Wright. The interpretation of population structure by f-statistics with special regard to systems of mating. Evolution, 19(3):395–420, 1965.

[64] CW Gini. Variability and mutability, contribution to the study of statistical distribution and relaitons. Studi Economico-Giuricici della R, 1912.

[65] Mark A Hall and Lloyd A Smith. Feature selection for machine learning: comparing a correlation-based filter approach to the wrapper. In FLAIRS conference, volume 1999, pages 235–239, 1999.

[66] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67(1):91–108, 2005.

[67] A Wayne Whitney. A direct method of nonparametric measurement selection. IEEE Transactions on Computers, 100(9):1100–1103, 1971.

[68] Thomas Marill and D Green. On the effectiveness of receptors in recognition systems. IEEE transactions on Information Theory, 9(1):11–17, 1963.

[69] Buse Çisil Otar and Süreyya Akyüz. Ensemble clustering selection by optimization of accuracy-diversity trade off. In Signal Processing and Communications Applications Conference (SIU), 2017 25th, pages 1–4. IEEE, 2017.

[70] Duygu Üçüncü, Süreyya Akyüz, Erdal Gül, and Gerhard Wilhelm-Weber. Optimality conditions for sparse quadratic optimization problem. In International Conference on Engineering Optimization, pages 766–777. Springer, 2018.

[71] Dimitris Bertsimas and Jack Dunn. Optimal classification trees. Machine Learning, 106(7):1039–1082, 2017.

[72] Shouqiang Du and Liping Zhang. A mixed integer programming approach to the tensor complementarity problem. Journal of Global Optimization, 73(4):789–800, 2019.

[73] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. Feature selection: A data perspective. ACM Computing Surveys (CSUR), 50(6):94, 2018.