Applied Soft Computing 113 (2021) 108006

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Ensemble learning via feature selection and multiple transformed subsets: Application to image classification

A. Khoder [b], F. Dornaika [a,b,c,*]

[a] Henan University, Kaifeng, China
[b] University of the Basque Country UPV/EHU, San Sebastian, Spain
[c] IKERBASQUE, Basque Foundation for Science, Bilbao, Spain

ARTICLE INFO

Article history:
Received 25 February 2021
Received in revised form 20 September 2021
Accepted 6 October 2021
Available online 1 November 2021

Keywords:
Ensemble learning
Feature subsets
Multi-models
Machine learning
Feature selection
Image classification
Class sparsity least square regression

ABSTRACT

In the machine learning field, especially in classification tasks, the model's design and construction are very important. Constructing the model via a limited set of features may sometimes bound the classification performance and lead to non-optimal performances that some algorithms can provide. To this end, Ensemble learning methods were proposed in the literature. These methods' main goal is to learn a set of models that provide features or predictions whose joint use could lead to a performance better than that obtained by the single model. In this paper, we propose three variants of a new efficient ensemble learning approach that was able to enhance the classification performance of a linear discriminant embedding method. As a case study we consider the efficient "Inter-class sparsity discriminative least square regression" method. We seek the estimation of an enhanced data representation. Instead of deploying multiple classifiers on top of the transformed features, we target the estimation of multiple extracted feature subsets obtained by multiple learned linear embeddings. These are associated with subsets of ranked original features. Multiple feature subsets were used for estimating the transformations. The derived extracted feature subsets were concatenated to form a single data representation vector that is used in the classification process. Many factors were studied and investigated in this paper including (Parameter combinations, number of models, different training percentages, feature selection methods combinations, etc.). Our proposed approach has been benchmarked on different image datasets of various sizes and types (faces, objects and scenes). The proposed scheme achieved competitive performance on four face image datasets (Extended Yale B, LFW-a, Gorgia and FEI) as well as on the COIL20 object dataset and the Outdoor Scene dataset. We measured the performance of our proposed schemes in comparison to (the single model ICS_DLSR, RDA_GD, RSLDA, PCE, LDE, LDA, SVM as well as the KNN algorithm) The conducted experiments showed that the proposed approach can enhance the classification performance in an efficient manner compared to the single-model based learning and was able to outperform its competing methods.

1. Introduction

Image classification is a widely investigated task in the machine learning and computer vision fields. Many researchers worked and focused on the implementation of both linear and non-linear models designed for classification tasks. Achieving reliable discriminative data representations is the objective in all the cases. It is a known fact that a more discriminative data representation will lead to enhanced classification performance. This is where the importance of engaging relevant data features in the model creation rises. Nowadays, representation learning is becoming more and more investigated [1–7]. Data features are usually separated into three categories, important (relevant), irrelevant or redundant. A good model should always target relevant features of the data and work on constructing the desired model using these features. This will ensure optimal classification performance.

Generally, specific features will ensure better representation for the data rather than other ones. These are referred to as relevant features. Authors in [8,9] has concluded that using the original data would not lead to the optimal classification performance in the learning applications. This should be addressed by extracting the most representative features from the original data. Data can then be analyzed via the extracted features. In addition to the problem that original data are not the best to work with, there exist another problem namely: curse of dimensionality, referring to the large number of features in the data. In real life and in specific applications, the dimension of the data can be

* Corresponding author at: University of the Basque Country UPV/EHU, San Sebastian, Spain.
E-mail address: fadi.dornaika@ehu.eus (F. Dornaika).

https://doi.org/10.1016/j.asoc.2021.108006

very large which makes their use very costly, both in time and computation wise. Various researchers focused on tackling this issue by using two main approaches namely: feature selection, and feature extraction. In these days, these schemes are highly targeted and play a major role in learning systems [10].

Researchers seek representation approaches that guarantee the delivery of a discriminative transformation matrix that has certain specifications and good discrimination abilities [11–14]. After that, one can use this transformation matrix to project the training and test data to the new derived space in order to obtain a new and more representative set of features. These features will be used in the construction of the model that will be then used in the classification tasks.

2. Literature

Most of the time single model based classifications were targeted and investigated. In other words, researchers work on proposing and implementing an algorithm in the purpose of achieving a good discriminative model that ensures good classification performance. Usually, in this process, what happens is that a model is created using the proposed algorithm, and then the output data is fed to a classifier for classification process to begin. In order to enhance the performance, one can use many known feature selection techniques (eg. Fisher score, ReliefF [15] and many more). Feature selection techniques have been widely used in the machine learning field [16]. In addition to that, one can perform a brutal search for the best features that are able to ensure the best classification performance provided by the proposed scheme, but still notice that the optimal performance was not achieved.

Recently, several optimization algorithms have been proposed. These newly proposed methods can also be considered or used as feature selection techniques. The authors in [17] have proposed an improved version of the whale optimization algorithm (WOA). The proposed algorithm is called "island-based whale optimization algorithm (iWOA)" and integrates the island model with the original (WOA). This mixture provided very good characteristics and resulted in optimized performance. Another recent optimization method is the exploratory cuckoo search (ECS). This proposed approach [18] incorporates three modifications to the original cuckoo search algorithm to enhance its exploration capabilities, and was also able to provide decent performance. Another method based on cuckoo search is the approach proposed by the authors in [19]. The authors in [19] proposed a memory-based cuckoo search algorithm as a feature selection technique. The proposed approach was able to store the most informative features identified by the best solutions using a memory-based mechanism, which helped to improve the classification performance. Another recently proposed optimization problem that can also be used for feature selection is the work presented by the authors in [20]. In the latter work, the authors proposed a hybrid optimization algorithm based on bitwise operations and Simulated Annealing to solve the Feature Selection problem for classification purposes using wrapper methods. The proposed method showed very good performance. Recently, many other optimization methods have been proposed which can also be used for feature selection [21].

In reality, it is not necessary that single model learning will always lead to the optimal performance provided by a proposed method even in the case of working with the most relevant features and applying feature selection techniques.

To address this issue, and investigate how to improve the performance of different methods, few researches talked about the ensemble learning methods. An Ensemble learning combines the predictions from multiple machine learning models into a single model which can reduce the generalization error. They offer increased flexibility and can scale in proportion to the amount of training data available. A couple of widely used ensemble approaches are bagging [22] and boosting [23].

The main idea of ensemble learning is to blend and combine the predictions from multiple models. These models are usually very good models and each one of them, taken separately, provides a good discriminant characteristic. By combining these models, one will obtain a single model that is described by its enhanced discrimination ability. Thus, leading to a better classification. So, the hypothesis is that in the case where the models are correctly combined, this can lead to more accurate and/or robust models. A variety of ensemble learning methods have been used in classification tasks mostly with deep convolutional neural networks (CNN's) for image classification. The reason is that ensemble learning has shown promising and excellent contribution in enhancing the performance of neural networks [24].

The performance of one single model is usually measured by its ability of obtaining the best predictor for the data. This can only be derived after the classification process finishes. There is no way to realize this information prior to that by only exploiting the handled data and the optimization problem [25]. This has been addressed in [25,26]. These researches focused on using a cross-validation strategy to evaluate the performance of each model individually. This strategy is referred to as the "discrete Super Learner selector".

One different view to ensure an enhanced performance can be the estimation of the optimal combination of the models that leads to the best predictor. This is well investigated in the literature. Brieman in [22] addressed and condensed several related works regarding the theoretical properties of ensemble learning [27–31]. Another well-known strategy used in ensemble learning is called "stacking" [32], it involves combining the predictions from multiple models on the same dataset. Many researchers have proposed linear combination approaches that introduced stacking to the ensemble of models [22,32].

In order to derive the most efficient combination of models, the work described in [22] investigated stacked regression by using cross-validation. The cross-validation based work has been expanded in the purpose of finding the best combination of predictors by proposing the "Super Learner" approach [25]. This framework demonstrated superiority and very good contributions in multiple areas namely: online learning [33], medicine [34, 35], spatial prediction applications [36] in addition to mortality prediction [37,38].

Over time, machine learning algorithms have been proposed for use in various fields (e.g., image processing [13,14,39], medical [40], predictive maintenance [41–43]). In recent years, several ensemble learning approaches have been proposed. These methods have led to remarkable performance improvements, and new optimization techniques have recently been published. Several multi-objective optimization methods have demonstrated their efficiency in various applications. An example of these methods is the "Evolutionary Ensemble Learning Using Multimodal Multi-objective Optimization Algorithm Based on Grid" presented by the authors in [44]. This method aims to improve the accuracy of wind speed forecasting for wind energy applications. Another recent ensemble learning approach was proposed by the authors in [45]. This proposed ensemble learning approach was proposed with the aim of addressing real-world applications, especially those where class imbalance is common. So, the proposed approach can be used in such cases to solve the class imbalance problem. Another interesting ensemble approach is the method proposed by the authors in [46], where the authors combined multiple artificial neural networks (ANNs) as a baseline (or weak learner) method for forecasting currency exchange rates. The authors in [47] proposed an ensemble learning based approach

to create parallel ensembles by applying the density peak criterion. The latter criterion works by generating different training sets, which leads to the generation of different classifiers and thus improves the classification performance. Another notable recent method based on ensembles is the method proposed by the authors in [48], where the authors propose a classification algorithm that uses multi-criteria optimization instead of relying on user-defined parameters. Although the authors claim that the method does not rely on user-defined parameters. However, in reality, the authors have used two user-defined parameters in their approach. But these parameters have a wide range which leads to statistically satisfactory results.

In this paper, we propose a novel framework used for supervised classification tasks. Instead of using an ensemble of classifiers, we propose the use of an ensemble of data representations. Our proposed approach is based on ensemble learning. The proposed approach creates multiple subsets of original features; these subsets are carefully chosen by using a single or multiple feature selection techniques. For each subset, a projection model (feature extraction) is built in order to get the transformed features. At the final stage, all transformed features are concatenated and used as a single large data representation that feed a classifier.

We make sure that the features of the data are ranked according to their importance by subjecting them to multiple feature selection techniques. In the way we have chosen to construct the features subsets, the most relevant features of the data were taken into consideration every time. Every created subset that we have used contains the most relevant features of the data overlapped with different features every time. In this way, even in the case where the chosen feature subset contains less relevant features, these features are there alongside with the most relevant ones and not alone. Moreover, due to the adopted feature ranking, the most relevant features will be used in several projection models.

The main idea of the proposed approach is generic and can be used by various methods. However, we have chosen the "Inter-class sparsity based discriminative least square regression" denoted as (ICS_DLSR) [14] as a backbone projection algorithm. This is motivated by (1) its remarkable discriminating ability, (2) efficient projection model computation, and (3) economic size of transformed features. The use of several feature selection techniques led to multiple variants of the proposed scheme. In brief, the paper has the following contributions:

- Proposing an ensemble of models based learning approach that improved the classification performance compared to single model learning.
- Studying the effect of the introduction of hybrid combination of multiple feature selection techniques into one single model.

The remainder of the paper is divided as follows: Section 3 will show the preliminaries. Section 4 is intended to describe the methodology of our proposed scheme. Section 5 will present the experimental results and method evaluation. Finally Section 6 concludes the paper.

3. Preliminaries

In current times, achieving an efficient data representation is the focus of many researches. Many studies are conducted for this purpose, and good methods have been delivered by various researchers [6,7,11,13,14]. To be able to test our ensemble learning based approach, we have chosen to use the " inter-class sparsity discriminative least square regression " (ICS_DLSR) [14] approach for multiple considerations. ICS_DLSR is an efficient method for both training and testing. It is flexible and has good discrimination properties. In this section, we will briefly describe some preliminaries. We will review the ICS_DLSR method and talk about the adopted feature selection techniques used for ranking the data features.

Table 1
Matrix norms.

Type	Formula
$\ell_{2,1}$ norm	$\|\mathbf{Z}\|_{2,1} = \sum_{i=1}^{C} \sqrt{\sum_{j=1}^{d} Z_{ij}^2}$
ℓ_F norm	$\|\mathbf{Z}\|_F = \sqrt{\sum_{i=1}^{C} \sum_{j=1}^{d} Z_{ij}^2}$

3.1. Notations

We proceed with the presentation of the notations used in our article. The training set is denoted as $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$, where d is the dimension of the samples. Each sample $\mathbf{x}_i$ is represented by a column vector consisting of 'd' features $\in \mathbb{R}^d$. N denotes the number of training samples. The total number of classes is denoted by C. The projection matrix is denoted as $\mathbf{Q} \in \mathbb{R}^{C \times d}$, and $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N] \in \mathbb{R}^{C \times d}$ is the label matrix corresponding to the training set $\mathbf{X}$, where each column vector $\mathbf{y}_i \in \mathbb{R}^C$ is simply defined as follows: If the training sample $\mathbf{x}_i$ belongs to the kth class, then the kth element of the column vector $\mathbf{y}_i$ is 1, while the remaining elements are 0.

Table 1 illustrates the computation of the $\ell_{2,1}$ and Frobenius norm (ℓ_F) for a matrix $\mathbf{Z} \in \mathbb{R}^{C \times d}$, where $\mathbf{Z}_{ij}$ denotes the (i, j)th element of the matrix $\mathbf{Z}$.

3.2. Review of Inter-class sparsity discriminative least square regression (ICS_DLSR) [14]

Original Least Square Regression (LSR) only focuses on fitting the input features to the corresponding output labels but still ignores the correlations among samples. LSR has been effective and proved very good contribution in many applications like gene classification [49], cancer classification [50], face recognition [51], image retrieval [52] and speech recognition [53].

Based on the LSR framework, the authors in [14] proposed the Inter-class sparsity discriminative least square regression (ICS_DLSR) method in order to obtain a more discriminative and compact projection space. This proposed framework imposed an inter-class sparsity constraint on the projected data which ensures that the derived projected data obtain common class structure. In addition, the authors introduced an error term with row-sparsity constraint to relax the strict zero–one label matrix. This allowed ICS_DLSR to be more flexible in the learning process. ICS_DLSR achieved superior performance and proved to be effective on many datasets. It aims to minimize the following problem:

$$\min_{\mathbf{Q},\mathbf{E}} \frac{1}{2}\|\mathbf{Y} + \mathbf{E} - \mathbf{Q}\mathbf{X}\|_F^2 + \frac{\lambda_1}{2}\|\mathbf{Q}\|_F^2 + \lambda_2 \sum_{i=1}^{C} \|\mathbf{Q}\mathbf{X}_i\|_{2,1} + \lambda_3 \|\mathbf{E}\|_{2,1} \quad (1)$$

In Eq. (1), $\mathbf{Q}$, $\mathbf{X}$ $\mathbf{E}$ and $\mathbf{Y}$ represent the linear transformation matrix, the data samples matrix, the error matrix, and the label matrix, respectively. λ_1, λ_2 and λ_3 are three parameters that determine the effect of the corresponding terms. C denotes the total number of classes. The matrix $\ell_{2,1}$ norm is used to promote the row-sparsity of a matrix. In this optimization problem, there are two unknown variables the linear transformation and the error matrix. To solve the problem, the authors adopted the alternating direction method of multipliers (ADMM) [54–56] to obtain the solution for $\mathbf{Q}$ and $\mathbf{E}$.

3.3. Feature selection techniques

In machine learning and computer vision, feature quality assessment is an important topic

In most of the learning problems, there exist hundreds or thousands of features describing each object. These features can either enhance the learning, or at particular occasions worsen it. For the purpose of ensuring the optimal learning performance, we should select the subset containing the most relevant features of the data. By doing so, one can enhance the performance and decrease the computational cost at the same time. Therefore, the problem of feature (attribute) selection has received much attention in the literature. Selecting the most relevant features of the data can be implemented using what is known by feature selection techniques.

- **Feature selection using Fisher score:**
  Generally, feature selection approaches main objective is selecting and highlighting the set of the relevant features of the original data. This selected subset of features is normally used to construct a more robust and compact model. Hence, leading to superior classification performance. Fisher score is one of the most famous algorithms used for feature selection, it works by computing the score of each data feature and then selects each feature accordingly.
  Fisher algorithm computes the score of the ith feature S_i by the following formula:

$$S_i = \frac{\sum_{j=1}^{C} n_j (\mu_{ij} - \mu_i)^2}{\sum_{j=1}^{C} n_j \rho_{ij}^2} \tag{2}$$

  where ρ_{ij} and μ_{ij} represent the variance and the mean of the ith feature associated with the jth class. The number of instances in the jth class is denoted by n_j and μ_i is the mean of the ith feature. C is the number of classes.
- **Feature selection using ReliefF score:**
  Original Relief Algorithm Another well-known algorithm that enables features ranking is the Relief algorithm. The majority of the approaches used for approximating the reliability of the attributes presume the conditional independence of the attributes and are thus less suitable for problems that might involve more feature interaction. Relief based algorithms (Relief, ReliefF and RReliefF) do not simply make this assumption [15,57,58].
  These algorithms are reliable, conscious of the contextual information, and can effectively estimate the quality and the relevance of attributes in problems with high attribute dependency. Relief algorithms are based on the concept of local margins for each feature. These margins should be large enough for relevant features. These algorithms are widely considered as feature subset selection methods used in the pre-processing phase before the model is trained [57]. They are still one of the most popular pre-processing algorithms to date [59]. They are actually general feature estimators which have been successfully used in a multitude of environments. Inspired by instance-based learning, the authors in [57] proposed the classical Relief algorithm. Relief is optimized for two-class problems. The basic principle of the algorithm is to consider not just the disparity in features values and the variance in the classes but also the distance between the instances.
  Let us consider the feature vector 'V' and the feature vectors of the instance closest to 'V' from each class. The closest instance belonging to the same group is referred to as near-hit (NH), and the closest instance with a different group is denoted as near-miss (NM).
  Relief Algorithm [15] iteratively computes the weight 'W' for the ith feature by:

$$W_i = W_i - (V_i - \mathrm{NH}_i)^2 + (V_i - \mathrm{NM}_i)^2 \tag{3}$$

  ReliefF Algorithm Authors in [15] improved the Relief algorithm. They developed an extension of the original Relief, called ReliefF, that improves the original algorithm by estimating margins more reliably. Irrelevant attributes either the redundant or noisy ones may affect the selection of the nearest neighbors. Thus, the estimation of the margins becomes unreliable. To address this problem, ReliefF searches for the "k" nearest (NH's) and (NM's) rather than a single (NH and NM) and averages the contribution of all k nearest (NH's) and (NM's). The selection of the nearest neighbors is very important in Relief-F. The purpose is to find the nearest neighbors with respect to important attributes. In all our experiments, "k" was set to 10 which, empirically, gives satisfactory results. In some problems significantly better results can be obtained in case of tuning "k" (as is typical for the majority of machine learning algorithms). Many studies were conducted to explore the feature selection ability using ReliefF algorithm [60]. More details about Relief variants can be found in [61].
- **Feature selection using Robust multi-label feature selection with dual-graph regularization:**
  The authors in the [62] have proposed a novel dual-graph regularization based feature selection method named "Robust multi-label feature selection with dual-graph regularization" (DRMFS). The proposed algorithm differs from the existing methods by incorporating only a single unknown variable (feature weight matrix) in its global criterion. Moreover, the developed approach is characterized by its ability to obtain a global optimal solution, unlike most competing methods with multiple unknown variables and their ability to obtain only local optimal solutions. DRMFS was developed based on the regularization of the feature graph and the regularization of the label graph. The former preserves the geometric structure of the features, while the latter considers the correlations of the data labels. The authors applied the $\ell_{2,1}$norm constraint to both the loss function and the weight matrix to improve the robustness of the method and ensure the row sparsity property. The objective function of the DRMFS algorithm is as follows:

$$\min_{\mathbf{W}} \|\mathbf{X}^T\mathbf{W} - \mathbf{Y}\|_{2,1} + \alpha\, Tr(\mathbf{W}^T\mathbf{L}^X\mathbf{W}) + \beta\, Tr(\mathbf{W}\mathbf{L}^Y\mathbf{W}^T) + \gamma\, \|\mathbf{W}\|_{2,1} \qquad s.t. \mathbf{W} \geq 0. \tag{4}$$

  where $\mathbf{X}$, $\mathbf{W}$, and $\mathbf{Y}$ denote the data, feature weight and label matrices, respectively. α, β and γ are three regularization parameters. $\mathbf{L}^X$ and $\mathbf{L}^Y$ represent the feature graph and label graph Laplacian matrices, respectively.
  Once the feature weight matrix $\mathbf{W}$ (the linear mapping) is computed, the score of each feature is given by $\|\mathbf{W}_{i*}\|_2$ ($1 \leq i \leq d$), where d denotes the dimensionality and $\mathbf{W}_{i*}$ is the ith row of $\mathbf{W}$. It is possible to retrieve the most relevant top-k features according to the highest scores ($k \leq d$). Detailed information about this proposed method is presented in [62].

4. Proposed ensemble class sparsity discriminative regression

In this section, we will describe our ensemble learning based approach. We will present the different phases of the process and the model construction.

4.1. Steps and methodology

Let us consider the data matrix $\mathbf{X} \in \mathbb{R}^{d\times N}$ where d and N represents the dimension (number of features) of the original data and the total number of samples, respectively. First, we apply one of the feature selection techniques over the original data.

- The score of each feature is computed (by one of the selection techniques stated above) and then features are ranked according to their scores. In this way, most relevant features, which are usually the ones with highest scores are placed at the top while the ones with lower scores are placed at the bottom. A graphical illustration of this weighting and ranking process is shown in Fig. 1. We denote the ranked features data matrix by $\mathbf{X}_s \in \mathbb{R}^{d\times N}$.
- Subsequent to the feature ranking process, we start by constructing our subsets of features. We construct multiple feature subsets in a way that each one is unique (coming from taking different percentages of features from the data matrix with ranked features) as it is shown in the upper part of Fig. 2. In its simplest implementation, the number of percentages defines the number of models, M. According to this scheme, the most relevant features of the data are taken into consideration in more than one subsets. Every created subset contains the most relevant features of the data overlapped with different features every time. Thus, even in the case where the chosen feature subset contains less relevant features, these features are there alongside with the most relevant ones and not alone. This ensures that no feature subset taken into consideration would harm the learning process.
- Let us consider creating M models. After generating the M subsets, the ICS_DLSR algorithm is applied on each subset that is fed as input data for the algorithm. In the ICS_DLSR algorithm process, each input generates a linear transformation matrix $\mathbf{Q}_n$ associated with this input. We have $n = 1, \ldots, M$.
- After obtaining the projection matrices $\mathbf{Q}_n$ delivered by ICS_DLSR, we can create our targeted data representations. We proceed by projecting each feature subset using the corresponding transformation $\mathbf{Q}_n$. Assuming that $\mathbf{X}$ represents the original data, after sorting according to the features scores this will be denoted as $\mathbf{X}_s$. Let $\mathbf{S}_n$ represents the data formed by the nth subset of features, $\mathbf{S}_n \subset \mathbf{X}_s$. It worth noting that the training and test data are submitted to the same procedure. Projecting training and test samples using $\mathbf{Q}_n$ is implemented by $\mathbf{A}_n = \mathbf{Q}_n\mathbf{S}_n$ and $\mathbf{B}_n = \mathbf{Q}_n\mathbf{T}_n$, where $\mathbf{S}_n$ corresponds to the training data formed by the $n-$th feature subset and $\mathbf{T}_n$ represents the test samples having the same subset of features. This leads to M models formed by the obtained descriptors (projected data vectors) with $n = 1, \ldots, M$.
- In the final stage of the proposed approach, the obtained M models are concatenated to form a single data representation which is finally fed to a given classifier (e.g., the Nearest Neighbor classifier). Since ICS_DLSR is used as a projection model, the dimension of the projection space provided by each model $\mathbf{Q}_n$ is C, the dimension of the final representation is $M \times C$.
  Fig. 2 depicts a graphical illustration of the main steps of the proposed approach. For simplicity, the case of **three models** creation was adopted in the example provided by this figure. This figure demonstrates the full process which includes: ranking the original features of the data, subsets construction, model creation, concatenation, and classification. Algorithmic steps of the proposed approach are illustrated in Algorithm 1.

Algorithm. 1. ICS_DLSR Based Ensemble Learning for Image Classification

Inputs:
1. Data samples $\mathbf{X} \in \mathbb{R}^{d\times N}$
2. Labels vector
3. Number of models, M
4. Percentages of subsets
5. Parameters λ_1, λ_2, λ_3
6. Feature selection technique

Steps:
1. Compute the scores and rank the features using one of the feature selection techniques (Fisher score, ReliefF, DRMFS, or other).
2. Select subsets of features according to the pre-defined percentages.
3. Apply the ICS_DLSR algorithm using each one of the extracted subsets of features as an input and derive the corresponding transformation matrices.
4. Project the training and test data on the new space using the obtained projection matrices associated with each input and construct the targeted models out of the transformed subsets.
5. Concatenate the obtained transformed subsets to form a single data representation vector.

Output: Data representation vector obtained by the concatenated models.

4.2. Proposed variants

We have proposed three variants of our approach namely: (i) Ensemble of models Class sparsity based discrimination using Fisher score EM_ICS_FS, (ii) Ensemble of models Class sparsity based discrimination using Combined score EM_ICS_HS and (iii) Ensemble of models Class sparsity based discrimination using the "Robust multi-label feature selection with dual-graph regularization" (DRMFS) algorithm [62] EM_ICS_DRMFS.

- Ensemble of models Class sparsity based discrimination using Fisher score **EM_ICS_FS**: In this variant of the approach, we have constructed a total of **10 models** in which the proportions of the data features taken from the original data are [10%, 20%, 30%, …,100%], respectively. The data contained in these models were obtained after original features are ranked via the **Fisher Score** feature selection technique only. The methodology of the model creation procedure is described in Fig. 2.
- Ensemble of models Class sparsity based discrimination using Combined score **EM_ICS_HS**: In this second variant, we have constructed a total of **10 models**. The main difference of this variant comes from the fact that the created models were obtained when the subsets of features were ranked using multiple feature techniques. In our experiments, **5 models** were created when the applied feature selection technique is the **Fisher Score** and the other **5 models** were constructed when we have applied **ReliefF** feature selection technique on the original data features. The proportions of the features taken from the data to construct the subsets for this variant are as follows [20%, 40%, 60%, 80%, 100%]. The methodology for the combined model creation is described in Fig. 3.
- Ensemble of models Class sparsity based discrimination using DRMFS algorithm **EM_ICS_DRMFS**: We have constructed a total of **10 models** in which the proportions of the data features taken from the original data are [10%, 20%, 30%, …,100%], respectively. The data contained in these models were obtained after original features are ranked via the recently proposed **DRMFS** algorithm.

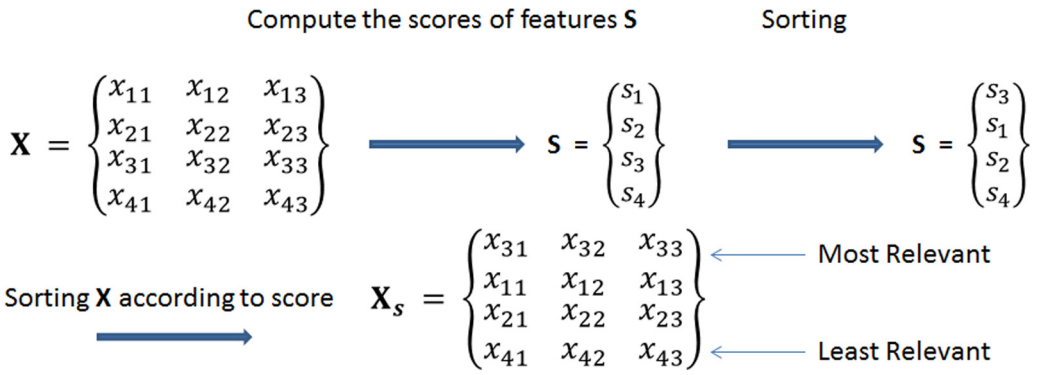


Fig. 1. Features ranking general methodology.

5. Experiments and analysis

5.1. Datasets

The datasets used in this paper are very well known in the field of computer vision, especially when working with classification tasks. Many other related works (proposing new image classification methods) have used some of these datasets for comparison. Moreover, we have used different image descriptors in our comparisons (Image rawbrightness, Local Binary Patterns and HOG features).

This section will provide detailed information regarding the datasets used in the experiments presented in this paper. Faces, objects and scene image datasets with different sizes were tested using our proposed approach.

- **Extended Yale B Face Dataset**[1]: The database used in this paper in the condensed version of the original Extended yale B dataset. Images in this dataset represent the faces of 38 different individuals while each one of these individuals has between 58 and 64 image. These face images were taken in various illuminations conditions and with different facial expressions for each person. A total number of 2414 images were used, each image is rescaled to 32×32 pixels. Raw brightness images of dimension 1024 are used in the experiments for this dataset. Results were derived while using different training percentages. 10, 15, 20, and 25 samples from each class were used as training samples and the remaining are used for testing.
- **LFW-a Dataset**[2]: "The Labeled Faces in the Wild-a (LFW-a)" is constructed from the images of the original LFW database after alignment using a commercial face alignment software. Images in this dataset maintained the same structure as in the original LFW dataset. This dataset contains a total of 3,408 image samples representing 141 classes. Raw brightness images of dimension 1,024 are used in the experiments. The reported results were obtained after we had varied the training percentage while using 5, 6, 7 and 8 image samples from each class as training samples. Remaining samples were used as test samples.
- **COIL20 Object Dataset**[3]: With the full name "The Columbia Object Image Library", COIL20 dataset contains images representing various objects. Each object is rotated around a vertical axis. It contains the images of 20 objects in which each object has 72 images, leading to a total number of 1,440 images. Local Binary Patterns (LBP) [63] are used as image descriptors in this dataset. We adopted the uniform LBP histogram (59 values). Three LBP descriptors are constructed from the image using 8 points and three values for the radius ($R = 1$, 2, and 3 pixels). As a result, the final concatenated descriptor has 177 values. We varied the training samples percentage, in our experiments we took 20, 25, 30, and 35 image samples from each class for training and the remaining were used as testing portions.
- **Georgia Face dataset**[4]: This dataset contains face images corresponding to 50 persons, each individual is represented by 15 images describing frontal and tilted faces with different facial expressions, lighting conditions and scale. The total number of images included in this dataset is 750 images. The images used are cropped and resized to 32×32 pixel for each image. Raw-brightness images of dimension 1024 are used in the experiments. The reported results are obtained after we used 3, 5, 7, and 9 image samples from each class as training samples and the remaining are used as test samples.
- **FEI dataset**[5]: The stated dataset contains pictures of the students and staff members at FEI. It is a face dataset that contains a set of colorful face images taken against a white background. The images are in an upright frontal position with profile rotation of up to about 180 degrees. This dataset contains a total number of 700 images, 14 images for each one of the 50 people. Raw brightness images of dimension 1024 are used. The reported results are obtained after we used 5, 6, 7, and 8 image samples from each class for training samples and the rest was used for testing.
- **Outdoor Scene dataset**[6]: This scenes dataset contains 2,688 images belonging to 8 groups. The descriptor used consists of 256 HOG features.

Table 2 presents a brief description of the datasets used in our paper, more information about these datasets can be found in the provided links presented in the footnotes. Fig. 4 shows some of the typical images included in the tested datasets.

5.2. Experimental setup

In the conducted experiments, the proposed approach is contrasted with many methods. We state from these: K-nearest neighbors (KNN) [64], Support Vector Machines (SVM) [65], Linear Discriminant Analysis (LDA) [66], Local Discriminant Embedding (LDE) [67], PCE [9], ICS_DLSR [14] and Robust sparse LDA (RSLDA) [13]. We note that the SVM used in the experiments is the **Linear SVM**, it was implemented using LIBSVM library.[7]

1 http://vision.ucsd.edu/~leekc/ExtYaleDatabase/ExtYaleB.html.
2 https://talhassner.github.io/home/projects/lfwa/index.html.
3 http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php.
4 http://www.anefian.com/research/face_reco.htm.
5 https://fei.edu.br/~cet/facedatabase.html.
6 https://github.com/sudalvxin/SMSC/tree/master/data.
7 https://www.csie.ntu.edu.tw/~cjlin/libsvm/.

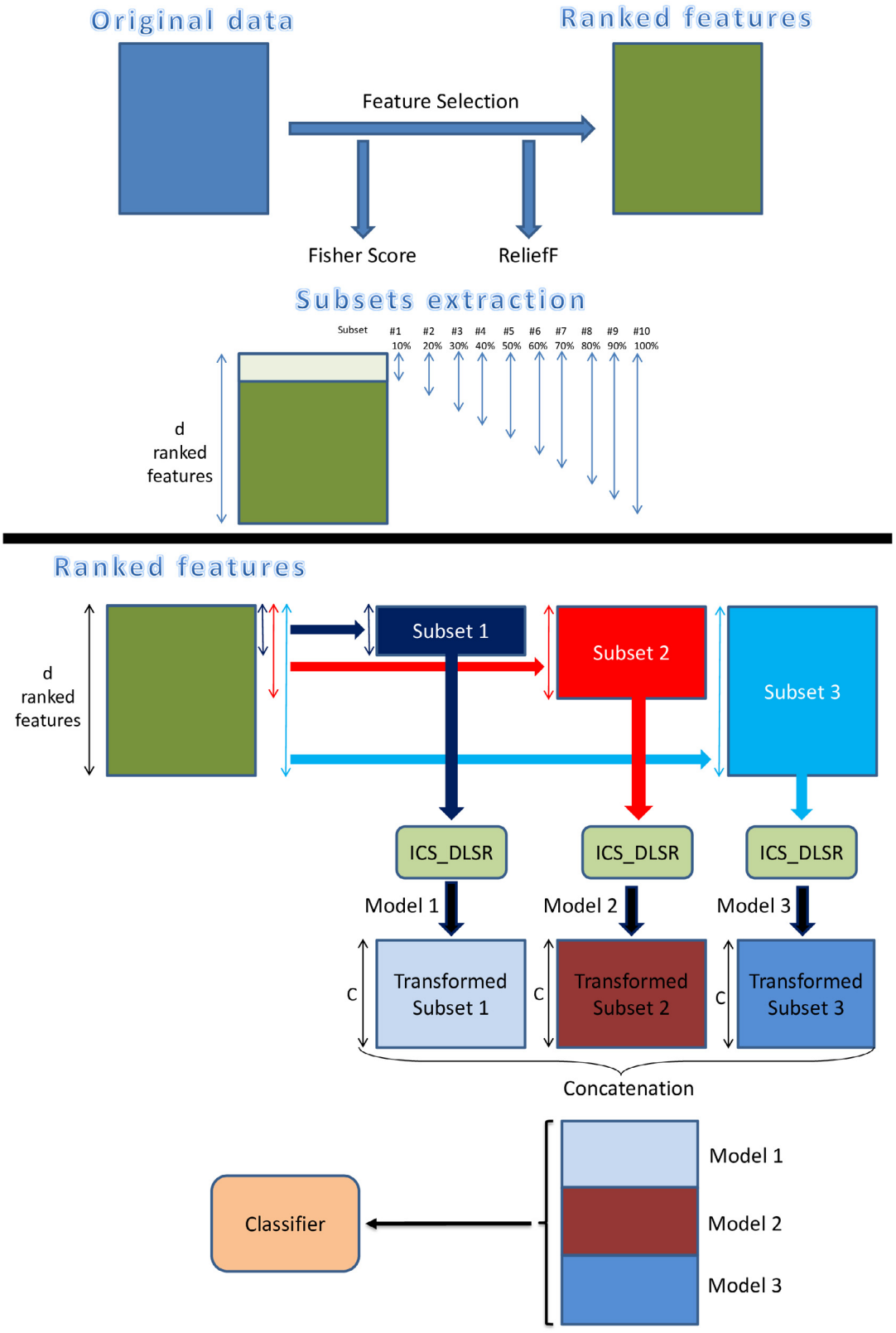


Fig. 2. Proposed ensemble learning methodology.

To further investigate the discrimination ability of the suggested approach, we have added some additional compared methods to the table of the Extended Yale B results 6. Robust Discriminant Analysis using Gradient Descent RDA_GD [39] , Linear Regression Based Classification (LRC) [68], Low-rank Linear Regression (LRLR) [69], Low-rank Ridge Regression (LRRR) [69], Sparse Low-rank Regression (SLRR) [69], Low-rank Preserving Projection via Graph Regularized Reconstruction (LRPP_GRR) [70] and Manifold Partition Discriminant Analysis (MPDA) [71] were added

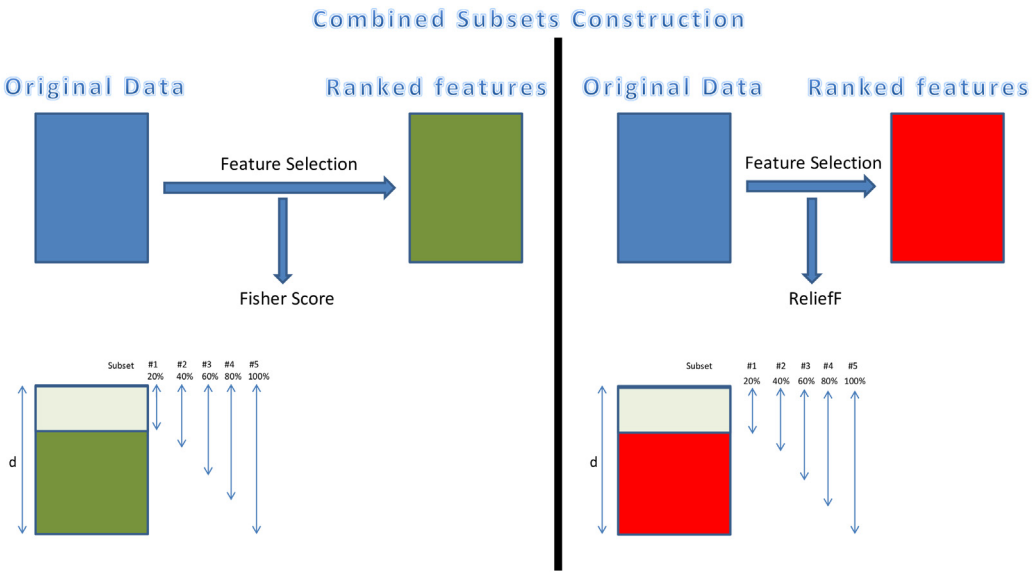
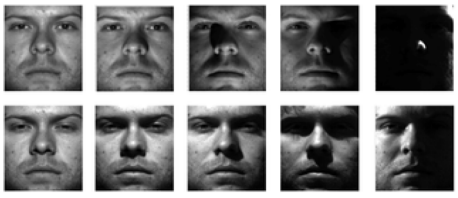


Fig. 3. Combined model construction methodology.

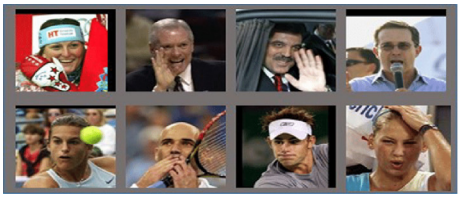
Table 2
Brief datasets description.

Dataset	Type	Number of samples	Number of features	Number of classes	Descriptor
Extended Yale B	Face	2414	1024	38	RAW-brightness images
LFW-a	Face	3408	1024	141	RAW-brightness images
COIL20	Object	1440	177	20	Local Binary Patterns
Georgia	Face	750	1024	50	RAW-brightness images
FEI	Face	700	1024	50	RAW-brightness images
Outdoor scene	Scene	2688	256	8	HOG features



(a) Images of the Extended Yale B dataset.

(b) Typical images of the COIL20 dataset.



(c) Typical images of the LFW-a dataset.

(d) Typical images of the Georgia dataset.

(e) Typical images of the FEI dataset.

Fig. 4. Typical images of various datasets.

to Table 6 in the purpose of widening the comparison among competing methods.

For a rational and accurate contrast, tests are carried out following the same experimental setup for all compared methods (eg, pre-processing and dimensionality reduction techniques). The classification performances presented in the tables are achieved using **10 splits** which were chosen **randomly** for each

dataset, unless specified otherwise in the table's caption. We report the average classification accuracy over the 10 splits.

In the conducted simulations, various training and test proportions were used for each dataset as detailed in Section 5.1. For each dataset and each compared approach, the targeted embedding matrix is first computed using the training data components. After that, the training and test data are projected onto the new space using the predicted embedding. And for the final step, classification of the test data is then performed using the Nearest Neighbor classifier (NN) [72]. The results presented in the tables were found with $K = 1$ (1-NN).

In our testing phase, we invoked dimensionality reduction of the raw features before feeding them to the learning models and classifiers most of the time. The Principal Component Analysis (PCA) was used as a pre-processing technique used for this purpose [73]. For the competing methods, PCA was used to preserve 100% of the data's energy. We note that, in some conducted experiments and for some methods e.g. (ICS_DLSR, in addition to the proposed approach), the original dimensionality was preserved and no pre-processing techniques were applied in order to highlight on the ability of the proposed approach in selecting the most relevant original features.

The reported classification rates of the methods are chosen from the best parameter configurations and correspond to the average over 10 randomly selected splits as mentioned before.

5.3. Experimental results

In this section, we will present the results derived through our experiments. We will compare our proposed method with the others mentioned in Section 5.2.

5.3.1. Feature selection techniques comparison

In this section, we study the performance of the proposed ensemble approach in the case of using three different feature selection methods to select the subsets of features that we are going to work with. Adopting multiple selection techniques have led to multiple variants of the proposed scheme. The main goal is to enhance the classification performance obtained by the original ICS_DLSR algorithm. In our experiments we have chosen the subsets of features that we are going to use after the original features have been ranked using Fisher score, a combination of ReliefF and Fisher score, in addition to ranking with the Robust multi-label feature selection with dual-graph regularization (DRMFS) [62] algorithm. The reason we have selected Fisher score and ReliefF feature selection techniques is that these algorithms have shown stability, very good performance and have been used widely in the machine learning field. We have also worked with the DRMFS algorithm in order to enrich the experiments.

The proposed variants denoted as **EM_ICS_FS** and **EM_ICS_DRMFS** represent our method where the features were ranked via the Fisher score and the DRMFS algorithm, respectively. The third variant denoted as **EM_ICS_HS** represents the case where the features were ranked via a hybrid combination using both ReliefF and fisher score algorithms.

Table 3 compares the classification performance of two variants of the proposed scheme alongside with the performance of the single model learning using the ICS_DLSR algorithm. Results presented in this table were obtained using the LFW-a dataset.

Table 4 presents the performance achieved by the proposed approach using two different feature selection algorithms. Classification rates presented in this table are obtained in case of using 10 models where the original data is ranked via the different algorithms. Results presented in this table were obtained using the COIL20 dataset.

Table 5 presents the classification performance obtained by the proposed variants compared to the performance associated with the single model ICS_DLSR algorithm over the Outdoor Scene dataset.

Table 3
Comparison of the mean classification performance (%) of different variants using LFW-a dataset.

LFW-a			
Training samples	Methods		
	ICS_DLSR	EM_ICS_FS	EM_ICS_HS
5	22.56	**27.38**	25.92
6	25.72	**31.75**	30.12
7	29.04	**36.07**	34.60
8	31.92	**39.71**	38.57

Table 4
Comparison of the mean classification performance (%) of different variants using the COIL20 dataset.

COIL20			
Training samples	Methods		
	ICS_DLSR	EM_ICS_FS	EM_ICS_DRMFS
20	98.04	98.36	**98.51**
25	98.22	98.61	**98.63**
30	98.75	98.92	**99.11**
35	99.12	99.21	**99.39**

Table 5
Comparison of the mean classification performance on the Outdoor Scene dataset.

Outdoor scene				
Training samples	Methods			
	ICS_DLSR	EM_ICS_FS	EM_ICS_HS	EM_ICS_DRMFS
50	68.19	68.75	**68.84**	68.80
70	69.41	**70.51**	70.15	70.11
90	69.64	**70.60**	70.41	70.45
110	70.21	71.03	**71.05**	70.78

5.3.2. Method comparison

Table 6 presents the classification performance of the proposed approach alongside with the competing methods using the first proposed variant over the Extended Yale B face dataset. Various training percentages were used. This table contains an extended number of compared methods, these methods were added to extend the comparison of the proposed method among other methods. Table 7 presents the obtained classification performance using the first proposed variant alongside with the competing methods over the LFW-a and COIL20 datasets.

Table 8 shows the obtained performance associated with two variants of the proposed scheme **EM_ICS_FS** and **EM_ICS_HS** next to the compared competing methods. Results presented in this table are noted over Georgia and FEI datasets.

The ensemble methodology presented in Figs. 2 and 3, which we used to build our models, allowed our proposed variants to outperform the competing methods. This is due to the fact that the most relevant features are always considered in all the models created, which provides powerful discrimination characteristics for each individual model and overall when concatenating the features of the models. The features presented in the created models were evaluated by various feature selection techniques and a hybrid scheme that exploits the powerful feature evaluation capability of several feature selection techniques simultaneously. This Methodology and the flexibility in choosing different feature selection techniques, allowed the different variants of our proposed scheme to outperform its competitors.

5.4. Parameters sensitivity analysis

This section's main objective is to describe and study the effect of the main parameters of our proposed approach. We will show

Table 6
Mean classification accuracies (%) of compared methods on the Extended Yale B dataset.

Ext. Yale B									
Training samples	Method	KNN	SVM	LDA	LDE	PCE	SULDA	RSLDA	RDA_GD
10		69.80	73.85	82.32	79.92	86.39	84.61	86.79	87.10
15		75.20	80.02	86.76	83.77	89.23	88.72	89.93	90.04
20		80.24	85.79	90.7	88.44	92.19	91.66	93.59	93.75
25		82.24	89.03	92.17	90.43	93.35	92.14	94.92	95.02
	Method	LRC	LRLR	LRRR	SLRR	LRPP_GRR	MPDA	ICS_DLSR	**EM_ICS_FS**
10		81.65	84.63	87.76	87.95	84.82	83.67	86.56	**88.46**
15		88.92	86.31	91.09	89.75	89.07	86.82	89.53	**91.43**
20		91.74	88.93	93.19	92.58	91.42	90.38	93.14	**94.49**
25		93.78	90.98	95.51	94.24	92.25	91.79	94.50	**95.88**

Table 7
Mean classification accuracies (%) of compared methods on the tested datasets using the first proposed variant **EM_ICS_FS**.

Dataset\Method	Training samples	KNN	SVM	LDA	LDE	PCE	RSLDA	RDA_GD	ICS_DLSR	EM_ICS_FS
LFW-a	5	9.90	12.72	20.51	9.98	9.44	24.70	25.11	22.56	**27.38**
	6	10.57	13.61	25.28	10.49	10.26	28.42	28.61	25.72	**31.75**
	7	11.06	14.70	28.62	11.24	10.98	31.50	31.82	29.04	**36.07**
	8	11.35	15.72	32.42	11.71	11.73	32.48	32.69	31.92	**39.71**
COIL20	20	94.58	97.65	96.19	95.00	94.87	96.73	96.89	98.04	**98.36**
	25	95.79	98.22	97.07	96.12	95.99	97.74	97.89	98.22	**98.61**
	30	96.65	98.70	97.81	97.01	97.49	98.26	98.52	98.75	**98.92**
	35	97.14	98.81	98.15	97.42	98.11	98.68	98.80	99.12	**99.21**

Table 8
Mean classification accuracies (%) of compared methods on the tested datasets using **EM_ICS_HS**.

Dataset\Method	Training samples	KNN	SVM	LDA	LDE	PCE	ICS_DLSR	EM_ICS_FS	EM_ICS_HS
Georgia	3	52.57	56.22	48.18	52.77	46.43	59.73	59.37	**59.95**
	5	61.28	66.98	59.20	62.14	56.18	71.12	71.40	**72.02**
	7	66.73	72.83	67.83	67.10	62.15	78.38	77.83	**79.03**
	9	71.40	77.53	72.57	72.13	66.37	82.57	81.93	**82.67**
FEI	5	88.98	91.18	92.60	90.67	86.04	92.16	92.20	**92.56**
	6	90.35	92.93	94.18	92.15	88.73	93.65	93.88	**94.20**
	7	92.60	94.31	95.60	94.26	91.09	95.20	95.14	**95.43**
	8	94.27	95.23	96.03	95.57	93.20	96.17	96.00	**96.27**

how the variation of the proposed approach's parameters affects the overall performance.

Like we have stated above, the ICS_DLSR algorithm minimizes the following objective function:

$$\min_{\mathbf{Q},\mathbf{E}} \frac{1}{2}\|\mathbf{Y}+\mathbf{E}-\mathbf{Q}\mathbf{X}\|_F^2+\frac{\lambda_1}{2}\|\mathbf{Q}\|_F^2+\lambda_2\sum_{i=1}^{C}\|\mathbf{Q}\mathbf{X}_i\|_{2,1}+\lambda_3\|\mathbf{E}\|_{2,1} \quad (5)$$

where $\mathbf{Q}$, $\mathbf{X}$ and $\mathbf{E}$ represent the transformation matrix, data samples and error matrix respectively. λ_1, λ_2 and λ_3 are three parameters to measure the effect of the corresponding terms. We have used the ICS_DLSR algorithm in our ensemble learning process. In our proposed approach, first we have selected multiple subsets of features using one or more feature selection techniques, then each subset of features was fed as an input to the ICS_DLSR algorithm to derive the associated transformation. Finally, we create the model out of the projected features.

Let us consider the subsets of features $\mathbf{Z}$, where $\mathbf{Z}_n \in \mathbb{R}^{m\times N}$ with $m \leq d$ represents the $n-$th features subset. $\mathbf{Z}_n^i$ denotes the $n-$th features subset corresponding to the $i-$th class. d and N denote the dimensionality of the data samples and the total number of the training data samples, respectively. Each feature subset is fed to the algorithm, our proposed approach work on minimizing the following problem:

$$\min_{\mathbf{Q},\mathbf{E}} \frac{1}{2}\|\mathbf{Y}+\mathbf{E}-\mathbf{Q}\mathbf{Z}_n\|_F^2+\frac{\lambda_1}{2}\|\mathbf{Q}\|_F^2+\lambda_2\sum_{i=1}^{C}\|\mathbf{Q}\mathbf{Z}_n^i\|_{2,1}+\lambda_3\|\mathbf{E}\|_{2,1} \quad (6)$$

According to experimental evaluations which we have conducted, we found that most of the time the optimal performance is obtained when the value of λ_3 is set to 1. Thus, we can set λ_3 to 1 and study the effect of changing the values of the two parameters λ_1 and λ_2 on the classification performance over different datasets. Fig. 5 presents the recognition rate as a function of the parameters using the original ICS_DLSR algorithm. Figs. 6 and 7 illustrate our findings, while using the first proposed scheme EM_ICS_FS and the second proposed scheme EM_ICS_HS, respectively.

Figs. 6 and 7 illustrate the variation of the classification performance obtained as a function of different parameter combinations using EM_ICS_FS and EM_ICS_HS. In general, our proposed method achieved satisfactory classification performance for a wide range of parameters used. For the tested datasets, the optimal performance was obtained when λ_1 and λ_2 are in the ranges $[1, 10^3]$ and $[1, 10^2]$, respectively (multiplicative factor is 10).

Table 9 briefly illustrates the parameter values that led to the optimal performance of our proposed method. This table presents the ranges where the parameters λ_1 and λ_2 achieved the best classification performance (multiplicative factor is 10), using different datasets and for the case where λ_3 is set to 1.

Another important factor in the ensemble learning, is the chosen number of created models, M, used for training. We have investigated about how the variation of the number of the created models affects the overall performance of the proposed scheme over the Extended Yale B dataset. Results presented in Fig. 8 are obtained while using 10 samples from each class from the Extended Yale B dataset for training and the remaining samples were used for testing.

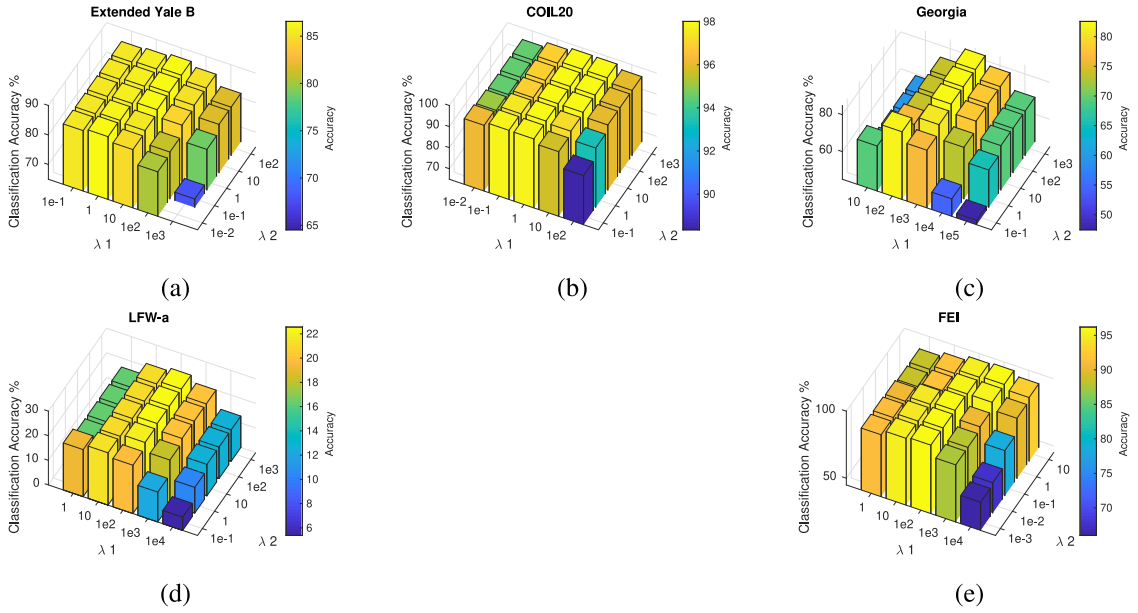


Fig. 5. Classification performance as a function of the parameters using the original ICS_DLSR method.

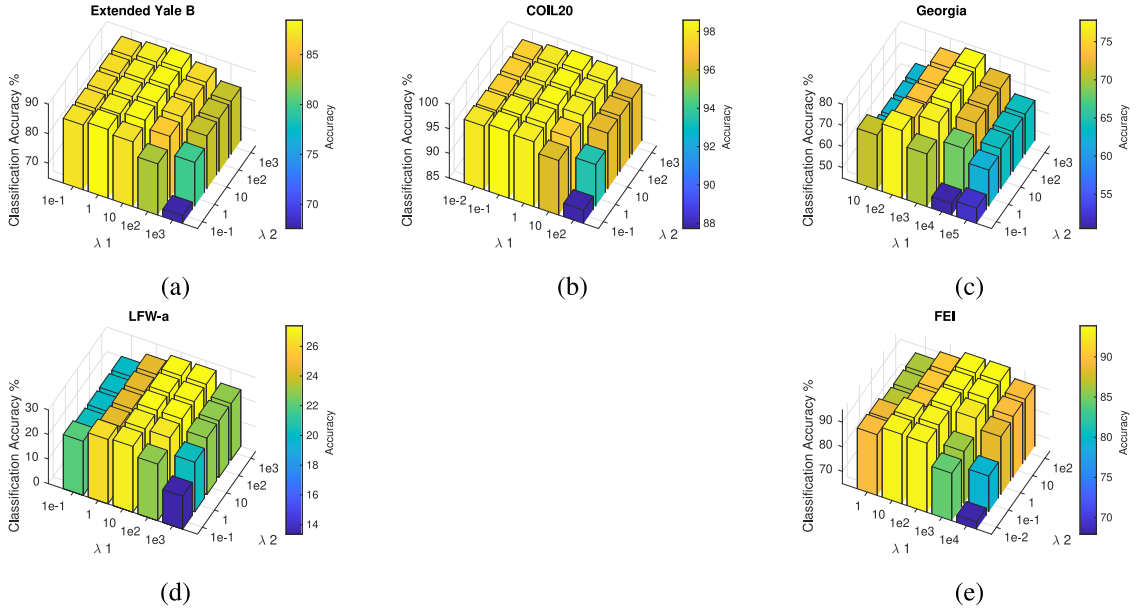


Fig. 6. Classification performance as a function of the parameters of the proposed method using **EM_ICS_FS**.

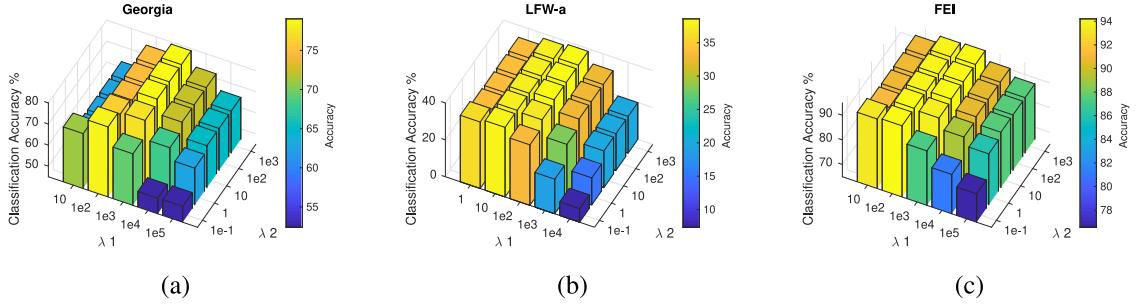


Fig. 7. Classification performance as a function of the parameters of the proposed method using **EM_ICS_HS**.

Table 9
Parameters sensitivity analysis of the proposed method vs. original ICS_DLSR.

Parameters sensitivity			
Method	Dataset	Optimal performance parameters range	
		λ_1	λ_2
ICS_DLSR	Extended Yale B	10	[1, 10]
	COIL20	1	$[10^{-3}, 10]$
	LFW-a	10^2	10
	Georgia	10^3	10
	FEI	$[10^2, 10^3]$	$[10^{-1}, 10]$
Proposed method	Extended Yale B	10	10
	COIL20	$[10^{-1}, 1]$	$[10^{-2}, 10]$
	LFW-a	$[10, 10^2]$	10
	Georgia	10^3	10
	FEI	$[10^2, 10^3]$	[1, 10]

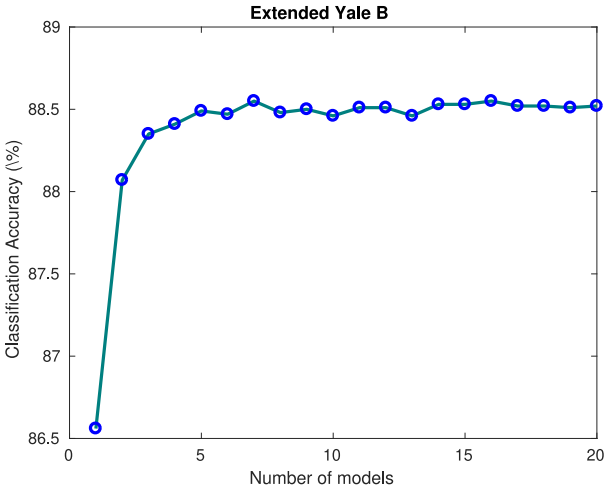


Fig. 8. Classification performance variation according to the number of models.

5.5. Analysis of the results

The experimental results illustrated in the previous figures and tables demonstrate the superiority of the suggested approach in comparison to other competing methods. Many observations can be made.

- The Proposed approach proved the superiority that ensemble learning can provide over single models. Conducted experiments have shown that by training multiple subsets of ranked features of original data, we can achieve better classification performance.
- The proposed variants were able to outperform the competing methods. This is due to the fact that the most relevant features are always taken into consideration in all of the constructed models, which provided powerful discrimination characteristics for each model separately and overall when the models are concatenated.
- We have proposed three variants for the proposed approach. All have shown very good discrimination properties and a remarkable enhancement over the baseline compared method, namely the ICS_DLSR method.
- For the datasets where the first variant of the proposed scheme failed to ensure an enhancement over the single model-based learning, other variants were able to enhance the classification performance and ensure the superiority of the proposed approach (e.g., the Georgia dataset using 3,7 and 9 training samples per class for training, and the FEI dataset when 7 and 8 training samples were used).
- The proposed approach is flexible in the sense that many other linear embedding approaches and feature selection techniques can be used and mixed to construct the desired models which may lead to a further better result.
- By analyzing the experimental results, we can observe that there is no specific feature selection technique that always leads to the best performance. The best option is to test multiple combinations to reach the optimal result. This in line with the literature of feature selection paradigms where the performance highly depends on the dataset used.
- Superior classification performance can be achieved if the parameters are accurately tuned. Very promising performance was obtained using a wide range for the used parameters, this is shown in Figs. 6 and 7.
- The studied ensemble learning approach can achieve noticeably better classification performance using a small number of models (refer to Fig. 8) and different training/testing portions of the data.
- The performance improvement brought by the proposed scheme with respect to the single model highly depends on the dataset used and the adopted feature ranking technique. For instance, on the Extended Yale B and LFW-a datasets, we obtained significant performance enhancement compared to the single model while using Fisher score as the feature ranking scheme. Fair classification improvement was also noted when using the Outdoor Scene dataset with the second proposed variant. For other datasets, less enhancement was observed using the ensemble learning.

6. Conclusion

In this paper, we have proposed three variants of an ensemble learning approach that have been able to enhance the classification performance of the class-sparsity based least-square regression (ICS_DLSR) method. Multiple feature subsets were used in the training process with the ICS_DLSR algorithm and their corresponding outputs were used to construct multiple models. These models are concatenated to form a single data representation which is used in the classification process. The targeted models were created by using various subsets of the original data. Our proposed approach's design ensures that each created model contains the most relevant features that describes the data efficiently. Relevant features are taken into consideration each time in a way that even if less relevant features are found they will not harm the classification performance. Original data features have been ranked using different and combined feature selection techniques. Many factors were studied and investigated in this paper including (parameter combinations, different number of models, different training percentages, hybrid methods combinations, etc.). The obtained findings proved that the proposed approach enhanced the classification performance compared to the single-model and was able to outperform competing methods. Our proposed approach has been benchmarked on different datasets and achieved competitive results. As with any other method, there are always some limitations to our proposed method. Multiple combinations using different feature selection techniques can fail to achieve the best hybrid feature combination to build the optimal models. Therefore, a trial of different combinations of feature selection techniques is sometimes required to construct the best models to be used later in the classification process.

CRediT authorship contribution statement

A. Khoder: Software, Validation, Resources, Investigation, Data curation, Writing – original draft, Writing – review & editing. **F. Dornaika:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] P. Langley, Selection of relevant features in machine learning: Defense Technical Information Center, 1994.
[2] Z. Li, J. Liu, Y. Yang, X. Zhou, H. Lu, Clustering-guided sparse structural learning for unsupervised feature selection, IEEE Trans. Knowl. Data Eng. 26 (9) (2013) 2138–2150.
[3] J.R. Quinlan, C4. 5: Programs for Machine Learning, Elsevier, 2014.
[4] L.E. Raileanu, K. Stoffel, Theoretical comparison between the gini index and information gain criteria, Ann. Math. Artif. Intell. 41 (1) (2004) 77–93.
[5] D. Wang, F. Nie, H. Huang, Feature selection via global redundancy minimization, IEEE Trans. Knowl. Data Eng. 27 (10) (2015) 2743–2755.
[6] S. Zang, Y. Cheng, X. Wang, J. Ma, Semi-supervised flexible joint distribution adaptation, in: Proceedings of the 2019 8th International Conference on Networks, Communication and Computing, 2019, pp. 19–27.
[7] Z. Zhao, X. He, D. Cai, L. Zhang, W. Ng, Y. Zhuang, Graph regularized feature selection with data reconstruction, IEEE Trans. Knowl. Data Eng. 28 (3) (2015) 689–700.
[8] N. Han, J. Wu, Y. Liang, X. Fang, W.K. Wong, S. Teng, Low-rank and sparse embedding for dimensionality reduction, Neural Netw. 108 (2018) 202–216.
[9] X. Peng, J. Lu, Z. Yi, R. Yan, Automatic subspace learning via principal coefficients embedding, IEEE Trans. Cybern. 47 (11) (2016) 3583–3596.
[10] N. Kwak, C.-H. Choi, Input feature selection for classification problems, IEEE Trans. Neural Netw. 13 (1) (2002) 143–159.
[11] F. Dornaika, A. Khoder, Linear embedding by joint robust discriminant analysis and inter-class sparsity, Neural Netw. (2020).
[12] A. Khoder, F. Dornaika, A hybrid discriminant embedding with feature selection: application to image categorization, Appl. Intell. (2020) 1–17.
[13] J. Wen, X. Fang, J. Cui, L. Fei, K. Yan, Y. Chen, Y. Xu, Robust sparse linear discriminant analysis, IEEE Trans. Circuits Syst. Video Technol. 29 (2) (2018) 390–403.
[14] J. Wen, Y. Xu, Z. Li, Z. Ma, Y. Xu, Inter-class sparsity based discriminative least square regression, Neural Netw. 102 (2018) 36–47.
[15] I. Kononenko, E. Šimec, M. Robnik-Šikonja, Overcoming the myopia of inductive learning algorithms with RELIEFF, Appl. Intell. 7 (1) (1997) 39–55.
[16] J. Cai, J. Luo, S. Wang, S. Yang, Feature selection in machine learning: A new perspective, Neurocomputing 300 (2018) 70–79.
[17] B.H. Abed-Alguni, A.F. Klaib, K.M. Nahar, Island-based whale optimisation algorithm for continuous optimisation problems, Int. J. Reason.-Based Intell. Syst. 11 (4) (2019) 319–329.
[18] B.H. Abed-alguni, N.A. Alawad, M. Barhoush, R. Hammad, Exploratory cuckoo search for solving single-objective optimization problems, Soft Comput. (2021) 1–14.
[19] M. Alzaqebah, K. Briki, N. Alrefai, S. Brini, S. Jawarneh, M.K. Alsmadi, R.M.A. Mohammad, I. ALmarashdeh, F.A. Alghamdi, N. Aldhafferi, et al., Memory based cuckoo search algorithm for feature selection of gene expression dataset, Inform. Med. Unlocked 24 (2021) 100572.
[20] M. Abdel-Basset, W. Ding, D. El-Shahat, A hybrid Harris Hawks optimization algorithm with simulated annealing for feature selection, Artif. Intell. Rev. 54 (1) (2021) 593–637.
[21] B.H. Abed-alguni, N.A. Alawad, Distributed Grey Wolf Optimizer for scheduling of workflow applications in cloud environments, Appl. Soft Comput. 102 (2021) 107113.
[22] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[23] H. Lu, R. Mazumder, Randomized gradient boosting machine, SIAM J. Optim. 30 (4) (2020) 2780–2808.
[24] L. Deng, J.C. Platt, Ensemble deep learning for speech recognition, in: Fifteenth Annual Conference of the International Speech Communication Association, 2014.
[25] M.J. van der Laanand, E.C. Polley, A.E. Hubbard, Super learner, Stat. Appl. Genet. Mol. Biol. 6 (2007).
[26] E.C. Polley, M.J. Van der Laan, Super learner in prediction, in: U.C. Berkeley Division of Biostatistics Working Paper Series. Working Paper 266, be Press, 2010.
[27] J.O. Berger, M. Bock, Combining independent normal mean estimation problems with unknown variances, Ann. Statist. (1976) 642–648.
[28] B. Efron, C. Morris, Combining possibly related estimation problems, J. R. Stat. Soc. Ser. B Stat. Methodol. 35 (3) (1973) 379–402.
[29] E.J. Green, W.E. Strawderman, A James-Stein type estimator for combining unbiased and possibly biased estimators, J. Amer. Statist. Assoc. 86 (416) (1991) 1001–1006.
[30] J. Rao, K. Subrahmaniam, Combining independent estimators and estimation in linear regression with unequal variances, Biometrics (1971) 971–990.
[31] D.B. Rubin, S. Weisberg, The variance of a linear combination of independent estimators using estimated weights, Biometrika 62 (3) (1975) 708–709.
[32] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259.
[33] D. Benkeser, C. Ju, S. Lendle, M. van der Laan, Online cross-validation-based ensemble learning, Stat. Med. 37 (2) (2018) 249–260.
[34] A.R. Luedtke, M.J. van der Laan, Super-learning of an optimal dynamic treatment rule, Int. J. Biostat. 12 (1) (2016) 305–332.
[35] R. Wyss, S. Schneeweiss, M. van der Laan, S.D. Lendle, C. Ju, J.M. Franklin, Using super learner prediction modeling to improve high-dimensional propensity score estimation, Epidemiology 29 (1) (2018) 96–106.
[36] M.M. Davies, M.J. Van Der Laan, Optimal spatial prediction using ensemble machine learning, Int. J. Biostat. 12 (1) (2016) 179–201.
[37] A. Chambaz, W. Zheng, M. Van Der Laan, Data-adaptive inference of the optimal treatment rule and its mean reward. the masked bandit, 2016.
[38] R. Pirracchio, M.L. Petersen, M. Carone, M.R. Rigon, S. Chevret, M.J. van der Laan, Mortality prediction in intensive care units with the Super ICU Learner Algorithm (SICULA): a population-based study, Lancet Respir. Med. 3 (1) (2015) 42–52.
[39] A. Khoder, F. Dornaika, An enhanced approach to the robust discriminant analysis and class sparsity based embedding, Neural Netw. (2021).
[40] T. Alsuliman, D. Humaidan, L. Sliman, Machine learning and artificial intelligence in the service of medicine: Necessity or potentiality? Curr. Res. Transl. Med. 68 (4) (2020) 245–251.
[41] Y.-H. Hung, Improved ensemble-learning algorithm for predictive maintenance in the manufacturing process, Appl. Sci. 11 (15) (2021) 6832.
[42] A. Mujib, T. Djatna, Ensemble learning for predictive maintenance on wafer stick machine using IoT sensor data, in: 2020 International Conference on Computer Science and its Application in Agriculture (ICOSICA), IEEE, 2020, pp. 1–5.
[43] S.K. Kiangala, Z. Wang, An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment, Mach. Learn. Appl. 4 (2021) 100024.
[44] Y. Hu, J. Liang, B. Qu, J. Wang, Y. Wang, P. Wei, Evolutionary ensemble learning using multimodal multi-objective optimization algorithm based on grid for wind speed forecasting, in: 2021 IEEE Congress on Evolutionary Computation (CEC), 2021, pp. 1727–1734.
[45] V.H.A. Ribeiro, G. Reynoso-Meza, Ensemble learning by means of a multi-objective optimization design approach for dealing with imbalanced data sets, Expert Syst. Appl. 147 (2020) 113232.
[46] L.T. Bui, T.T.H. Dinh, et al., A novel evolutionary multi-objective ensemble learning approach for forecasting currency exchange rates, Data Knowl. Eng. 114 (2018) 40–66.
[47] S. Roshan, S. Asadi, Development of ensemble learning classification with density peak decomposition-based evolutionary multi-objective optimization, Int. J. Mach. Learn. Cybern. 12 (6) (2021) 1737–1751.
[48] S. Fletcher, B. Verma, M. Zhang, A non-specialized ensemble classifier using multi-objective optimization, Neurocomputing 409 (2020) 93–102.
[49] Y. Li, A. Ngom, Nonnegative least-squares methods for the classification of high-dimensional biological data, IEEE/ACM Trans. Comput. Biol. Bioinform. 10 (2) (2013) 447–456.
[50] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1–3) (2002) 389–422.
[51] Y. Xu, X. Fang, Q. Zhu, Y. Chen, J. You, H. Liu, Modified minimum squared error algorithm for robust classification and face recognition experiments, Neurocomputing 135 (2014) 253–261.
[52] Q. Feng, Y. Zhou, R. Lan, Pairwise linear regression classification for image set retrieval, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4865–4872.
[53] D. Kim, M. Gales, Noisy constrained maximum-likelihood linear regression for noise-robust speech recognition, IEEE Trans. Audio Speech Lang. Process. 19 (2) (2010) 315–325.
[54] Z. Lin, M. Chen, Y. Ma, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, 2010, arXiv preprint arXiv:1009.5055.
[55] Z. Lin, R. Liu, Z. Su, Linearized alternating direction method with adaptive penalty for low-rank representation, 2011, arXiv preprint arXiv:1109.0367.
[56] J. Yang, X. Yuan, Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization, Math. Comp. 82 (281) (2013) 301–329.
[57] K. Kira, L.A. Rendell, A practical approach to feature selection, in: Machine Learning Proceedings 1992, Elsevier, 1992, pp. 249–256.
[58] I. Kononenko, M.R. Šikonja, Non-myopic feature quality evaluation with (R) ReliefF, in: Computational Methods of Feature Selection, Chapman and Hall/CRC Press, Goshen, CT, 2008, pp. 169–191.

[59] T.G. Dietterich, Machine-learning research, AI Mag. 18 (4) (1997) 97–136.
[60] M. Robnik-Šikonja, I. Kononenko, Theoretical and empirical analysis of ReliefF and RReliefF, Mach. Learn. 53 (1–2) (2003) 23–69.
[61] S. Hijazi, Semi-Supervised Margin-Based Feature Selection for Classification (Ph.D. thesis), Université du Littoral Côte d'Opale; Université Libanaise, école doctorale, 2019.
[62] J. Hu, Y. Li, W. Gao, P. Zhang, Robust multi-label feature selection with dual-graph regularization, Knowl.-Based Syst. 203 (2020) 106126.
[63] L. Li, P.W. Fieguth, G. Kuang, Generalized local binary patterns for texture classification., in: BMVC, Vol. 123, 2011, pp. 1–11.
[64] L. Kozma, k Nearest Neighbors Algorithm (kNN), Helsinki University of Technology, 2008.
[65] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (3) (2011) 27.
[66] A. Tharwat, T. Gaber, A. Ibrahim, A.E. Hassanien, Linear discriminant analysis: A detailed tutorial, AI Commun. 30 (2) (2017) 169–190.
[67] H.-T. Chen, H.-W. Chang, T.-L. Liu, Local discriminant embedding and its variants, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 2, IEEE, 2005, pp. 846–853.
[68] I. Naseem, R. Togneri, M. Bennamoun, Linear regression for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (11) (2010) 2106–2112.
[69] X. Cai, C. Ding, F. Nie, H. Huang, On the equivalent of low-rank linear regressions and linear discriminant analysis based regressions, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, pp. 1124–1132.
[70] J. Wen, N. Han, X. Fang, L. Fei, K. Yan, S. Zhan, Low-rank preserving projection via graph regularized reconstruction, IEEE Trans. Cybern. 49 (4) (Apr. 2019) 1279–1291.
[71] Y. Zhou, S. Sun, Manifold partition discriminant analysis, IEEE Trans. Cybern. 47 (4) (2016) 830–840.
[72] P. Cunningham, S.J. Delany, K-nearest neighbour classifiers, Mult. Classif. Syst. 34 (8) (2007) 1–17.
[73] L.I. Smith, A Tutorial on Principal Components Analysis, Technical report, 2002.