Multiobjective Sparse Ensemble Learning by Means of Evolutionary Algorithms

Jiaqi Zhao[a], Licheng Jiao[b], Shixiong Xia[a,*], Vitor Basto Fernandes[c,d], Iryna Yevseyeva[e], Yong Zhou[a], Michael T. M. Emmerich[f]

[a]*School of Computer Science and Technology, China University of Mining and Technology, No 1, Daxue Road, Xuzhou, Jiangsu, 221116, China*

[b]*Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, International Research Center for Intelligent Perception and Computation, Joint International Research Laboratory of Intelligent Perception and Computation, Xidian University, Xi'an Shaanxi Province 710071, China*

[c]*Instituto Universitário de Lisboa (ISCTE-IUL), University Institute of Lisbon, ISTAR-IUL, Av. das Forças Armadas, 1649-026 Lisboa, Portugal*

[d]*School of Technology and Management, Computer Science and Communications Research Centre, Polytechnic Institute of Leiria, 2411-901 Leiria, Portugal*

[e]*Faculty of Technology, De Montfort University, Gateway House 5.33, The Gateway, LE1 9BH Leicester, UK*

[f]*Multicriteria Optimization, Design, and Analytics Group, LIACS, Leiden University, Niels Bohrweg 1, 2333-CA Leiden, The Netherlands*

Abstract

Ensemble learning can improve the performance of individual classifiers by combining their decisions. The sparseness of ensemble learning has attracted much attention in recent years. In this paper, a novel multiobjective sparse ensemble learning (MOSEL) model is proposed. Firstly, to describe the ensemble classifiers more precisely the detection error trade-off (DET) curve is taken into consideration. The sparsity ratio (*sr*) is treated as the third objective to be minimized, in addition to false positive rate (*fpr*) and false negative rate (*fnr*) minimization. The MOSEL turns out to be augmented DET (ADET) convex hull maximization problem. Secondly, several evolutionary multiobjective algorithms are exploited to find sparse ensemble classifiers with strong performance. The relationship between the sparsity and the performance of ensemble classifiers on the ADET space is explained. Thirdly, an adaptive MOSEL classifiers selection method is designed to select the most suitable ensemble classifiers for a given dataset. The proposed MOSEL method is applied to well-known MNIST datasets and a real-world remote sensing image change detection problem, and several datasets are used to test the performance of the method on this problem. Experimental results based on both MNIST datasets and remote sensing image change detection show that MOSEL performs significantly better than conventional ensemble learning methods.

Keywords: Ensemble Learning, sparse representation, classification, multiobjective optimization, change detection.

1. Introduction

The idea of ensemble learning methods [1] is to construct a set of classifiers with base learning algorithms and then classify new data points by taking a (weighted) vote of their predictions. Generally, ensemble methods combine the prediction of individual methods and can obtain better predictive performance than any individual method alone. Ensemble learning methods have attracted much attention in recent years. Not only have many ensemble algorithms been proposed [2, 3], but also ensemble learning methods have been applied to many areas [4, 5], such as medical information processing [1] and satellite image classification [6].

In general, an ensemble learning algorithm is constructed in two steps, i.e., training a number of component classifiers and then combining the predictions of the components. The most prevailing approaches for training component classifiers are bagging [7], boosting [8], random subspace [9], and rotation forest [10]. Recently, research has drawn attention to multiobjective optimization of ensemble learning [11, 12] and several evolutionary multiobjective algorithms (EMOAs) have been used to deal with it. Generally, most of this work is trying to obtain a set of classifiers with good performance on both diversity and accuracy by using multiobjective optimization algorithms with different objectives. The multiobjective deep belief networks (DBNs) ensemble method was proposed in [13], in which an MOEA was applied to evolve multiple DBNs by considering accuracy and diversity as two conflicting objectives. A divide-and-conquer based optimization framework for ensemble classifiers generation was proposed in [12], in which the accuracy of each class was treated as the objectives to describe the performance of classifiers. Besides, maximizing the ensemble size is also taken as an additional objective. The Pareto image features were applied for candidate classifiers generation in [14] by using a multiobjective evolutionary trace transform algorithm. These methods do not consider the redundancy between classifiers and the efficiency of ensemble learning, as it requires a significant amount of memory to store the candidates of classifiers and lots of computation time is also needed to predict the label of each new input instance.

In this paper, we focus on combining the predictions of component classifiers by finding several appropriate sparse weight vectors for them. Many works have addressed the complexity of ensemble classifiers by reducing the number of classifiers in the component candidate set. The relationship between the en-

*Corresponding author. Tel.: +86 051683591709.
Email address: shixiongxia.cumt@outlook.com (Shixiong Xia)

semble learning and its component classifiers is analyzed in [15], which reveals that a better performance can be obtained by ensembling many instead of all the available classifiers. A genetic algorithm is adopted to evolve the weights of the component classifiers, showing that it can generate ensemble classifiers with small sizes but good generalization ability. The theoretical and empirical evidence in [16] suggests that a smaller ensemble size can often obtain better performance than a larger ensemble. It is, therefore, possible to obtain an ensemble which minimizes the number of individual classifiers and preserves or improves the performance of attributes, such as accuracy and cost of misclassification. However, only the accuracy is considered in this method, the result contains redundant classifiers, as the sparsity of ensemble classifiers is not considered. Several pruning strategies are analyzed in [17], including reduction error (RE), Kappa pruning (KP), complementarity measure (CM) and margin distance (MD). Matching pursuit (MP) is used to prune the ensemble classifiers in [18] by balancing the diversity and the individual accuracy. In these methods, the greedy strategy is used to search for the optimal classifiers set and it is easy to fall into the local extremum.

Sparse ensembles were proposed in [19]. The outputs of multiple classifiers were combined by using a sparse weight vector. The *hinge loss* and the *1-norm* regularization were exploited to calculate the sparse weight vector, formulated as a linear programming problem. However, the *1-norm* metric cannot describe the sparseness of ensemble classifiers precisely. This is because a weight vector with a group of small values can improve the performance of *1-norm* measurement but cannot improve the performance of sparseness. The *0-norm* metric can describe the sparseness more precisely [20]. The sparse ensemble learning is applied for synthetic aperture radar (SAR) image classification in [6] and for Youtube videos classification in [21]. The *0-norm* learning can be regarded as an NP-hard problem, it is still an open problem to search the global optimum.

Compressed sensing (CS) [22] was brought to ensemble learning in [23]. It explores the globally optimal subset of classifiers for a given ensemble. To solve the compressed sensing problem, a sparse weighting vector which contains many zeros should be generated first, and then appropriate weights should be provided for the remaining classifiers according to their relative importance. Several popular methods such as SpaRAS [24], OMP [25], FISTA [26], PFP [27] are used to tune the weight vector of ensemble classifiers. In [23] it is shown that compressed sensing ensembles are often as accurate as, or more accurate than, conventional ensembles, although they use only small subsets of the total set of classifiers. However, the sparseness

should be set in advance when using the compressed sensing methods. Meanwhile, the characteristics of the unbalanced data classification were not taken into consideration.

The contributions and drawbacks of the most related works of literature are listed in Table 1. Above all, the drawbacks of these methods are listed in the following: 1) The optimization algorithms used were easily trapped into local extremum; 2) Only accuracy metric cannot describe ensemble performance precisely; 3) The relationship between sparsity ensemble weights and ensemble performance was not analyzed in depth.

Table 1: Contributions of several important literatures

Literature	Contributions	Drawbacks
Zhou et al. (2002) [15]	The relationship between the ensemble learning and its component classifiers is analyzed.	The sparsity of ensemble was not taken into consideration.
Martínez-Munõz et al. (2009) [17]	Several pruning strategies were proposed.	The strategy is easy to fall into the local extremum.
Chen et al. (2009) [16]	Theoretical suggests that a smaller ensemble size can often obtain better performance than a larger ensemble.	The strategy is easy to fall into the local extremum.
Mao et al. (2011) [18]	Matching pursuit (MP) is used to prune the ensemble classifiers by balancing the diversity and the individual accuracy.	The strategy is easy to fall into the local extremum.
Zhang et al. (2011) [19]	The *hinge loss* and the *1-norm* regularization was exploited.	The *1-norm* metric cannot describe the sparseness of ensemble classifiers precisely.
Li et al. (2014) [23]	A compressed sensing approach for efficient ensemble learning.	The sparseness should be set in advance.

In this paper, we propose the novel concept of a multiobjective sparse ensemble learning (MOSEL) method, in which the relationship between the sparsity and the classification performance is explained. To accurately describe the performance of ensemble classifiers, the detection error trade-off (DET) [28] performance is taken into consideration by adopting the false positive rate (*fpr*) and the false negative rate (*fnr*) simultaneously. Besides, the sparsity ratio (*sr*) of ensemble classifiers is treated as the third objective to be minimized. The DET can describe the classifiers more precisely than the accuracy metric especially for unbalance data classification problems [28]. Besides, the evolutionary multiobjective algorithm (EMOA) [29] technique is first applied to evolve the combining weights of ensemble component classifiers. With the technique of tri-objective ensemble learning, we can obtain a set of ensemble classifiers with different sparseness, rather than an ensemble classifier with a certain sparseness that is previously set. The sparsity and the error rates of ensemble classifiers are explainable, and their trade-offs are quantifiable in the augmented DET (ADET) space.

We analyze the properties of the ADET for sparse ensemble learning and several state-of-the-art many-objective optimization algorithms are applied to solve multiobjective ADCH maximization problems, in-

cluding the two-archive algorithm (Two_Arch2) [30], which focuses on convergence and diversity separately, the decomposition based algorithms, such as NSGA-III [31], the evolutionary algorithms based on both dominance and decomposition (MOEA/DD) [32], the reference vector guided evolutionary algorithm (RVEA) [33], an indicator based evolutionary algorithm with a reference point adaptation (AR-MOEA) [34], and 3D convex-hull-based evolutionary multiobjective optimization algorithm (3DFCH-EMOA) [35, 36]. By using EMOAs, we can obtain a set of potentially optimal ensemble classifiers with different *sr-fpr-fnr* trade-offs.

The remaining paper is organized as follows. Section 2 gives a brief introduction to multiobjective optimization of a sparse ensemble method. Section 3 presents the results of several classification problems with MNIST [37] and remote sensing change detection datasets, and Section 4 provides concluding remarks.

2. Multiobjective sparse ensemble learning

2.1. Ensemble Learning

The idea of a *sparse ensemble* of classifiers is to combine the predictions of all classifiers in the candidate set using a sparse weight vector. The sparse vector has many elements with the value of zero and only classifiers corresponding to nonzero weights are selected for the ensemble. To improve the performance of the ensemble classifier and to reduce the memory demand for the components, it is required to select an optimal subset of classifiers and the corresponding weights vector for this subset. The problem of seeking sparse weights vectors can be modeled as a combinatorial optimization problem, which can be solved by evolutionary algorithms [30].

In this paper, we only consider binary supervised ensemble classification problems. With a set of training samples $X_{tr} = \{(x_j, y_j)|x_j \in R^d, y_j \in \{-1, +1\}, j = 1, 2, \ldots, M_{tr}\}$, where y_j is the class label corresponding to a given input x_j, d is the dimensionality of sample of features, and M_{tr} is the number of instances. Note that in this work we only consider binary classification problems and we set the labels as $\{-1, 1\}$, where 1 represents positive category and -1 represents negative category, given a set of classifiers $\{C_1(x), C_2(x), \ldots, C_N(x)\}$, where $C_i(x)$ is the i-th classifier in the candidate ensemble set. Usually, the classifier $C_i(x)$ is obtained by using the training dataset X_{tr} with the strategy of random selection of the features or the instances.

A classifier can be obtained by using a training dataset with a machine learning algorithm, which can be described as an estimate of the unknown function $y = f(x)$. The classifier $C_i(x)$ is a hypothesis $f_i(x)$ about the true function $f(x)$, which can predict the class label y for a new input vector x from a testing dataset X_{ts} or a validation dataset X_{val}. Usually, the training dataset is used for base classifiers learning, the validation dataset is used for ensemble pruning, and the test dataset is used for ensemble classification performance evaluation. Denote by f_{ji} the prediction of the ith learner $C_i(x)$ for the jth sampling of the validation sample x_j, that is described by Eq. (1).

$$f_{ji} = C_i(x_j). \tag{1}$$

The prediction output label vector $\mathbf{f_i}$ can be obtained by implementing the classifier C_i for the validation dataset X_{val} with size M_{val}, which is denoted as in Eq. (2).

$$\mathbf{f_i} = [f_{1i}, f_{2i}, \ldots, f_{M_{val}i}]^T. \tag{2}$$

The matrix $\mathbf{F}$ of prediction labels for all instances obtained by all of the classifiers can be denoted by Eq. (3),

$$\mathbf{F} = [\mathbf{f_1}, \mathbf{f_2}, \ldots, \mathbf{f_N}] \tag{3}$$

where $\mathbf{f_i} = [f_{1i}, f_{2i}, \ldots, f_{Mi}]^T$, $i = 1, 2, \ldots, N$, and $\mathbf{F} \in R^{M_{val} \times N}$.

The ensemble learning can improve the performance of classifiers by combining the decisions of each classifier and assigning weight w_i to each of the classifier $C_i(x)$, and the vector of weights $\mathbf{w}$ is denoted by Eq. (4).

$$\mathbf{w} = [w_1, w_2, \ldots, w_N]^T. \tag{4}$$

The predicted label vector $\mathbf{y}_{predict}$ obtained by ensemble learning for the input dataset X can be described as in Eq. (5).

$$\mathbf{y}_{predict} = \mathbf{F}\mathbf{w}. \tag{5}$$

The perfect ensemble classifier can be obtained by solving an equation $\mathbf{y}_{val} = \mathbf{y}_{predict}$. Usually, the number of equations is larger than that of the weighting variables in the equation system. In this case, there are typically no exact solutions for equations. In this case, the equation system can be approximately solved by using optimization algorithms to find solutions, which can minimize the difference between the training labels and predicting labels.

2.2. Multiobjective optimization of ensemble learning

The DET curve [28] is taken into consideration to describe the performance of ensemble classifiers, which has been proved to be a good measurement to evaluate the performance of classifiers [38]. The definition of the DET curve is closely related to the two-by-two confusion matrix, which describes the relationship between the ground truth and the predicted class for a binary classifier. A confusion matrix is shown in Table 2, which includes four possible outcomes. An outcome is a *true positive* if a positive instance is correctly classified and it is a *true negative* if a negative instance is correctly classified. Whenever a negative instance is classified as positive, we call it a *false positive*. Finally, whenever a positive instance is classified as negative, we call it a *false negative*.

Table 2: A two-by-two confusion matrix of binary classifiers

		True class	
		P^+	N^-
Predicted class	P^+	True positives (TP)	False positives (FP)
	N^-	False negatives (FN)	True negatives (TN)

Let TN denote the number of true negatives, FP the number of false positives, TP the number of true positives, and FN the number of false negatives. Then the *false positive rate* (*fpr*) is defined as $fpr = \mathrm{FP}/(\mathrm{TN} + \mathrm{FP})$, and the *false negative rate* (*fnr*) is defined as $fnr = \mathrm{FN}/(\mathrm{TP} + \mathrm{FN})$. To minimize the difference between true labels and predicted labels, both *fpr* and *fnr* should be minimized.

To obtain sparse ensemble classifiers with good performance, not only should the difference between true label vector $\mathbf{y}_{val}$ and predicted label vector $\mathbf{y}_{predict}$ be minimized, but also the number of nonzero elements in the weight vector $\mathbf{w}$ should be minimized. In Eq. (6) we define the sparsity ratio (*sr*) to describe the sparseness of ensemble,

$$sr = \frac{\|\mathbf{w}\|_0}{N}. \tag{6}$$

Here, N is the number of classifiers in the candidate ensemble set and $\|\mathbf{w}\|_0$ represents the number of nonzero entities in the weight vector. The weight vector $\mathbf{w}$ is constrained to non-negative values, as negative weightings are neither intuitively meaningful nor reliable [23]. We try to find ensemble classifiers with a low value of *sr* in order to reduce classification effort and to counteract overfitting of the ensemble classifier.

The computational cost of an ensemble classifier with high *sr* is considered to be higher than that of an ensemble classifier with lower *sr*. We prefer an ensemble classifier with lower *sr* when given two ensemble classifiers with the same performance criteria (*fpr*, *fnr*). So *sr*, *fpr* and *fnr* are conflicting with each other. A low value of *sr* means that a small number of classifiers are selected for the ensemble, i.e., the ensemble classifier has a low value of *sr*, which would result in a poor performance of *fpr* and *fnr*. By treating the sparse term *sr* as the third objective, the sparse ensemble turns out to be a multiobjective problem. We denote it as multiobjective sparse ensemble learning (MOSEL) which is described in Eq. (7),

$$\begin{aligned} \min \mathrm{MOSEL}(\mathbf{w}) &:= (fpr, fnr, sr)(\mathbf{w}), \\ \text{subject} \quad \text{to} \quad & \mathbf{w} \in \Omega, \end{aligned} \tag{7}$$

where Ω is the set of all possible weight vectors and $\mathbf{w}$ refers to the weightings with good performance of sparse ensemble classifiers.

2.3. Sparse real encoding

The sparse real encoding strategy is designed to represent the weight vector for the evolutionary algorithms, which is an improved version of the real encoding method. The sparse real encoding is constituted by an array of real values in the interval [0, 0.1]. The length of the chromosome is determined by the number of candidate classifiers for ensembles. Two strategies are used to modify the real encoding approach for multiobjective sparse ensembles. One is called *hard threshold sparse* the other is called *inequality constraint*. Details will be discussed below.

The classifier with a small value of weight in the ensemble learning system does not contribute much to the final decision. In this paper, we ignore the classifiers with small values by adopting a hard threshold

strategy. The value of weights smaller than the threshold is set to zero, as described in Eq. (8)

$$\mathbf{w}_{update}(i) = \begin{cases} 0, & \text{if } \mathbf{w}(i) < \sigma \\ \mathbf{w}(i), & \text{else,} \end{cases}$$

where σ is the hard threshold. In the experimental section, the value is set to 0.05, where N is the number of candidate classifiers. The sparse real encoding can model the solution of sparse ensemble learning, and then several EMOAs can be applied to evolve the individuals in the population set.

2.4. Adaptive MOSEL classifiers selection

The proposed MOSEL can deliver a set of ensemble classifiers, in this part we designed an adaptive selection method to choose the most suitable classifier for a given dataset [39]. Let $p(P^+)$ signify the frequency of positive samples and $p(N^-)$ denote that of negative samples for a dataset. With an ensemble classifier, the risk (R) can be denoted as Eq. 8,

$$R = \lambda(FN, P^+) \cdot p(P^+) \cdot fnr + \lambda(FP, N^-) \cdot p(N^-) \cdot fpr, \tag{8}$$

where $\lambda(FN, P^+)$ is the loss incurred for deciding *Negative* when the true label is *Positive* and so is $\lambda(FP, N^-)$. In many real-world problems we can not obtain the label of each sample, however, we can estimate the distributions of a dataset with a predefined classifier, and we denote them as $\hat{p}(P^+)$ and $\hat{p}(N^-)$. Specifically, we do not consider cost-sensitive classification problem in this paper, Eq. 8 can be simplified as Eq. 9:

$$R = \hat{p}(P^+) \cdot fnr + \hat{p}(N^-) \cdot fpr. \tag{9}$$

The most suitable ensemble classifier can be selected by minimizing the risk R. The adaptive MOSEL classifiers selection algorithm is described in Alg. 1. Firstly, randomly select an ensemble classifier from the *mosel* set, and then evaluate the distributions of the given dataset. Under the evaluated distributions we can select the most suitable ensemble classifier by minimizing Eq. 9. If the selected classifier is the same as the preselected one it can be returned as the most suitable classifier, else go back to Step 2.

Algorithm 1 Adaptive MOSEL classifiers selection ($mosel$, X_{ts})

Require: $mosel$ is the ensemble classifiers set, the performance of each classifier in ADET space with X_{val} can be obtained

Ensure: the most suitable ensemble classifier for X_{ts}

1: Set $t \leftarrow 0$ and select a classifier i_t from the solution set $mosel$ randomly
2: Predict the labels for X_{ts} by $EnC_{wi_t}$and evaluate the dataset distributions $\hat{p}_t(P^+)$ and $\hat{p}_t(N^-)$
3: $t \leftarrow t + 1$
4: $i_t \leftarrow \arg\min_{j=1}^{n} \hat{p}_{t-1}(P^+) \cdot fnr_j + \hat{p}_{t-1}(N^-) \cdot fpr_j$
5: **if** $i_t = i_{t-1}$ **then**
6: return EnC_{i_t}
7: **else**
8: Go to step 2
9: **end if**

2.5. Framework of MOSEL

The description of the framework of MOSEL is given in Alg. 2. Firstly, we train a set of candidate classifiers with X_{tr} by adopting bagging or random subspace strategies. Secondly, optimize the sparse vector $\mathbf{w}$ by using EMOAs with X_{val}, which is used to evaluate the performance of each individual of the EMOAs. Thirdly, the most suitable ensemble classifiers for X_{ts} can be obtained by adopting adaptive MOSEL classifiers selection algorithm.

Algorithm 2 Learning Procedure for MOSEL

1: Training a set of candidate of classifiers with X_{tr}
2: Optimizing the sparse vector $\mathbf{w}$ by using EMOAs with X_{val}
3: Obtain the most suitable ensemble classifier for X_{ts} by using adaptive MOSEL classifiers selection algorithm

3. Experimental studies

3.1. Algorithms involved

In this section, we present the experimental results of the proposed multiobjective sparse ensemble learning methods and then compare the results with the results obtained by two compressed sensing (CS) ensemble methods and two pruning ensemble methods. The sparse ensemble methods in our comparison include SpaRAS [24], OMP [25], which are the most popular methods for solving sparse reconstruction problems [23]. The compared pruning methods are Kappa pruning (KP) [17] and ensemble based on matching pursuit (MP) [18]. Several state-of-the-art EMOAs are used to search the solutions of MOSEL, including

Two_Arch2 [30], NSGA-III [31], MOEA/DD [32], RVEA [33], AR-MOEA [34] and 3DFCH-EMOA [36]. The MNIST [37] and remote sensing change detection datasets are selected to evaluate the performance of the above methods. The strategy of random subspaces [9] is adopted as the dataset manipulation and the classification and regression tree (CART) [40] is used as the base learner. For each mentioned algorithm, 10 independent trials are conducted.

3.2. Parameter setting

The experiment stopping criteria of the six EMOAs are set with a maximum of 30000 function evaluations. The simulated binary crossover (SBX) and polynomial bit-flip mutation operators are applied in the experiments with crossover probability of $p_c = 0.9$ and the mutation probability of $p_m = 0.1$. The population size is set to 100 for all EMOAs. All of the experiments were implemented using Matlab code running on an IBM X3650 server with Xeon E5-2600 2.9GHz processors and 32GB memory under Ubuntu 16.04. The details of experiments are described in the following.

3.3. Metrics

Six metrics are chosen to evaluate the performance of studied algorithms in the comparative experiment on these datasets, including *Accuracy*, *Precision*, *Recall*, *F-measure*, Kappa coefficient (*Kappa*) [41] and number of non-zero ensemble weight. Generally, *Accuracy*, *Precision*, *Recall* and *F-measure* are popular in the area of binary classification problem. The definition of them is denoted in Eq. 10. The larger the value of them the better is the classification performance.

$$\begin{aligned} Accuracy &= \frac{TP+TN}{TP+TN+FP+FN}, \\ Precision &= \frac{TP}{TP+FP}, \\ Recall &= \frac{TP}{TP+FN}, \\ F-measure &= \frac{2\times Precision \times Recall}{Precision + Recall} \end{aligned} \tag{10}$$

Kappa is a statistic indicator which measures inter-rater agreement for categorical items. It is generally thought to be a more robust measure than simple percent agreement calculation, as *Kappa* takes into account

the possibility of the agreement occurring by chance. The definition of *Kappa* is denoted in Eq. 11.

$$\begin{aligned} Kappa &= \frac{Accuracy - p_e}{1 - p_e}, \\ p_e &= \frac{TP \times (TP + FN) + TN \times (TN + FP)}{TP + TN + FP + FN}. \end{aligned} \tag{11}$$

Generally, the larger the value of the *Kappa*, the better performance of the algorithm. The number of non-zero ensemble weight is selected to describe the sparse ratio of ensemble weight. We prefer a low value of this metric. The statistical results of these metrics are listed in the following tables. In these tables the best results obtained are marked in light grey and the second best results are marked in dark grey. Furthermore, the Wilcoxon sum-rank test [36], which is a statistical test, is selected to evaluate whether the differences between 3DFCH-EMOA (one of MOSEL methods) and other methods are significant or not.

3.4. Experimental results on MNIST datasets

3.4.1. Dataset description

The MNIST dataset [37] is widely used for machine learning and pattern recognition methods on real-world data. It contains a training set with 60,000 examples and a testing set with 10,000 examples. Some samples from MNIST dataset are shown in Fig. 1. The handwritten digits have been size-normalized and centered in a fixed-size image (i.e., 28×28). The intensity of each pixel in an image is treated as its features, so the dimensionality of features set for each sample is 784. In this part, we use a small amount of examples for training and validation, and the remains for testing.

Figure 1: Samples from MNIST dataset

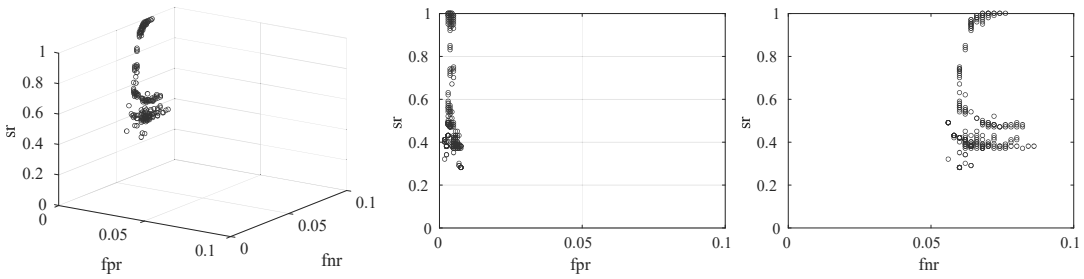
Table 3: The details of MNIST dataset used in the experiments

class	No. all set	No. of testing	No. of training	ds1	ds2	ds3	ds4	d5	ds6	ds7	d8	ds9
0	6903	5923	980	+	-	-	-	-	-	-	-	-
1	7877	6742	1135	-	+	-	-	-	-	-	-	-
2	6990	5958	1032		-	+	-	-	-	-	-	-
3	7141	6131	1010			-	+	-	-	-	-	-
4	6824	5842	985				-	+	-	-	-	-
5	6313	5421	892					-	+	-	-	-
6	6876	5918	958						-	+	-	-
7	7293	6265	1028							-	+	-
8	6825	5851	974								-	+
9	6958	5949	1009									-

The MNIST dataset we used in this part is described in the left part of Table 3. As we only consider binary classification problems in this paper, we select several sub-datasets from the whole dataset, including ds1-ds9 (details are listed in the right part of Table 3). All of the sub-datasets contain two classes, for instance, the positive class in ds2 includes '1', and the negative class includes '0' and '2'. Both balanced and unbalanced datasets are created; for instance, in the ds9 dataset, the ratio of positive instances to negative instances is about 1:9. For each of the datasets, 1/2 of training instances are randomly selected for candidate classifiers generation, and the rest is used for ensemble performance evaluation.

3.4.2. Experimental results and discussion

Firstly, the reference Pareto front is shown to illustrate the properties of solutions of tested EMOAs, which is calculated as the best set of solutions of several algorithms achieved in the first experimental run. Without loss of generality, we only discuss the result of the ds3 dataset in Table 3.

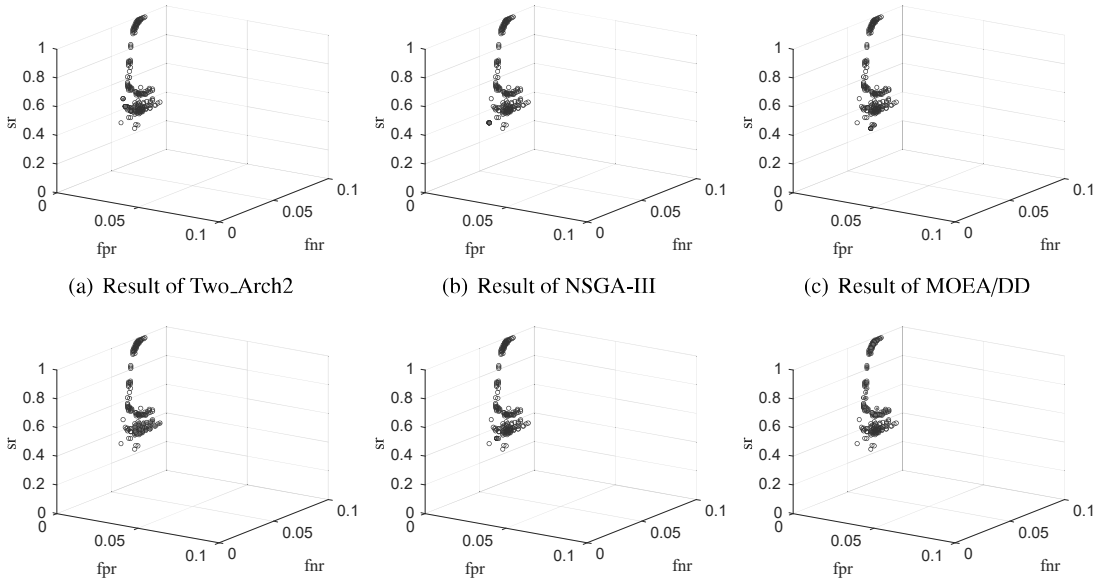


(a) The reference Pareto front in 3D (b) *(fpr* × *sr)* projection (c) *(fnr* × *sr)* projection

Figure 2: The reference Pareto front for ds3 dataset

The obtained reference Pareto front is shown in Fig. 2(a). We can see that the reference Pareto front includes a set of discrete points on the ADET surface. To illustrate the reference Pareto front clearly, two

dimensional projections are shown in Fig. 2(b) and Fig. 2(c), corresponding to $fpr \times sr$ projection and $fnr \times sr$ projection, respectively. From Fig. 2(b) we conclude that: 1) The *fpr* could not be reduced to zero, even with all of the classifiers active, but it got very close to it; 2) The best result of *fpr* can be obtained with the value of *sr* in the range of $[0.3, 0.75]$ and in the range of $[0.75, 1.0]$, which is almost exactly zero; 3) There are no points (solutions) in the objective space region with the value of *sr* below 0.3, as the performance of the *fpr* is too bad. From Fig. 2(c) we conclude that: 1) The performance of *fnr* decreases with the decreasing of *sr*, when *sr* is above 0.8; 2) The best result of *fnr* is obtained with the value of *sr* in the range of $[0.3, 0.5]$; 3) The performance of *fnr* is suppressed when the value of *sr* is below 0.3. Taking the conclusions of Fig. 2 together, some more conclusions can be made: 1) The *fpr*, *fnr* and *sr* are conflicting with each other, as they cannot reach the best result simultaneously; 2) The highest value of *sr* can not guarantee the best performance of *fpr* and *fnr*; 3) Very few classifiers can reduce the performance of ensemble learning, as the performance of both *fpr* and *fnr* degrades when the value of *sr* is lower than 0.3. The solutions of each EMOA are discussed next.



(a) Result of Two_Arch2 (b) Result of NSGA-III (c) Result of MOEA/DD

(d) Result of RVEA (e) Result of AR-MOEA (f) Result of 3DFCH-EMOA

Figure 3: The Pareto front for ds3 dataset (three axis projection) obtained by six EMOAs

The Pareto front and reference Pareto front by six EMOAs are shown in Fig 3, in which the points of the Pareto front are marked in red and the points of reference Pareto front are marked in blue. By Comparing

all Pareto front in Fig 3, we can see that: 1) The solutions of Two_Arch2, NSGA-III, MOEA/DD, and AR-MOEA convergence to the local area; 2) The solutions of RVEA and 3DFCH-EMOA are distributed in a wider space; 3) RVEA can find solutions with low value of *sr*; 4) 3DFCH-EMOA can obtain solutions with a high and low value of *sr*.

Table 4: Mean and standard deviation of *Accuracy* of ensemble methods on MNIST datasets

Methods \ Datasets	ds1	ds2	ds3	ds4	ds5
MP	0.9889 ± 0.0028	0.9766 ± 0.0023	0.9355 ± 0.0051	0.9419 ± 0.0046	0.9570 ± 0.0035
KP	0.9829 ± 0.0038	0.9695 ± 0.0043	0.9204 ± 0.0046	0.9232 ± 0.0076	0.9503 ± 0.0051
SpaRSA	0.9921 ± 0.0054	0.9836 ± 0.0082	0.9583 ± 0.0201	0.9506 ± 0.0187	0.9741 ± 0.0132
OMP	0.9885 ± 0.0032	0.9765 ± 0.0024	0.9347 ± 0.0043	0.9418 ± 0.0046	0.9561 ± 0.0035
Two_Arch2	0.9957 ± 0.0006	0.9891 ± 0.0007	0.9704 ± 0.0016	0.9641 ± 0.0014	0.9787 ± 0.0016
NSGA-III	0.9952 ± 0.0007	0.9893 ± 0.0005	0.9696 ± 0.0021	0.9651 ± 0.0011	0.9798 ± 0.0012
MOEA/DD	0.9952 ± 0.0012	0.9893 ± 0.0009	0.9698 ± 0.0022	0.9644 ± 0.0018	0.9797 ± 0.0010
RVEA	0.9959 ± 0.0008	0.9886 ± 0.0007	0.9696 ± 0.0016	0.9639 ± 0.0015	0.9778 ± 0.0018
AR-MOEA	0.9959 ± 0.0009	0.9897 ± 0.0004	0.9702 ± 0.0015	0.9648 ± 0.0018	0.9795 ± 0.0009
3DFCH-EMOA	0.9962 ± 0.0005	0.9894 ± 0.0005	0.9707 ± 0.0018	0.9654 ± 0.0012	0.9802 ± 0.0013

Methods \ Datasets	ds6	ds7	ds8	ds9	Average
MP	0.9484 ± 0.0041	0.9714 ± 0.0023	0.9724 ± 0.0018	0.9472 ± 0.0024	0.9599
KP	0.9345 ± 0.0062	0.9622 ± 0.0048	0.9650 ± 0.0048	0.9416 ± 0.0036	0.9500
SpaRSA	0.9624 ± 0.0102	0.9748 ± 0.0123	0.9796 ± 0.0066	0.9578 ± 0.0052	0.9704
OMP	0.9488 ± 0.0043	0.9714 ± 0.0022	0.9726 ± 0.0016	0.9472 ± 0.0024	0.9597
Two_Arch2	0.9614 ± 0.0034	0.9830 ± 0.0008	0.9823 ± 0.0006	0.9569 ± 0.0016	0.9757
NSGA-III	0.9638 ± 0.0025	0.9828 ± 0.0006	0.9827 ± 0.0006	0.9579 ± 0.0015	0.9762
MOEA/DD	0.9630 ± 0.0022	0.9829 ± 0.0003	0.9823 ± 0.0007	0.9568 ± 0.0013	0.9759
RVEA	0.9604 ± 0.0020	0.9825 ± 0.0006	0.9820 ± 0.0005	0.9556 ± 0.0013	0.9751
AR-MOEA	0.9619 ± 0.0026	0.9828 ± 0.0006	0.9822 ± 0.0005	0.9569 ± 0.0013	0.9760
3DFCH-EMOA	0.9630 ± 0.0021	0.9832 ± 0.0006	0.9827 ± 0.0004	0.9569 ± 0.0016	0.9764

Table 4 shows the mean and standard deviation of *Accuracy*. The average classification accuracy for each method is listed in the last column of the table. By comparing all the results, we can conclude that the methods of MOSEL outperform CS and pruning ensemble methods. 3DFCH-EMOA and NSGA-III outperform other methods for most of the datasets.

The statistical results of *Kappa* are shown in Table 5. By comparing the results on the table, we can see that MOSEL methods outperform CS and pruning methods for most of the MNIST datasets. NSGA-III and 3DFCH-EMOA outperform other methods on most of these datasets. NSGA-III plays slightly better than 3DFCH-EMOA in the metric of *Kappa*. SpaRSA performs better than other CS and pruning ensemble methods.

As most of the datasets used in this part are large and the distributions of them are unbalanced, a small improvement of the accuracy and *Kappa* can cause many samples to be correctly classified and reduce misclassification costs greatly. To show the classification performance in more detail, the *Precision*, *Recall*

Table 5: Mean and standard deviation of *Kappa* of ensemble methods on MNIST datasets

Methods \ Datasets	ds1	ds2	ds3	ds4	ds5
MP	0.9776 ± 0.0056	0.9493 ± 0.0050	0.8214 ± 0.0129	0.8089 ± 0.0159	0.8377 ± 0.0145
KP	0.9657 ± 0.0077	0.9338 ± 0.0092	0.7765 ± 0.0147	0.7474 ± 0.0233	0.8123 ± 0.0184
SpaRSA	0.9841 ± 0.0108	0.9644 ± 0.0177	0.8844 ± 0.0550	0.8395 ± 0.0588	0.9022 ± 0.0491
OMP	0.9769 ± 0.0064	0.9492 ± 0.0051	0.8193 ± 0.0109	0.8088 ± 0.0158	0.8342 ± 0.0136
Two_Arch2	0.9914 ± 0.0012	0.9763 ± 0.0016	0.9174 ± 0.0045	0.8805 ± 0.0050	0.9183 ± 0.0065
NSGA-III	0.9903 ± 0.0015	0.9767 ± 0.0012	0.9151 ± 0.0061	0.8844 ± 0.0037	0.9228 ± 0.0047
MOEA/DD	0.9904 ± 0.0024	0.9767 ± 0.0020	0.9159 ± 0.0063	0.8818 ± 0.0062	0.9222 ± 0.0042
RVEA	0.9917 ± 0.0016	0.9751 ± 0.0015	0.9150 ± 0.0046	0.8798 ± 0.0055	0.9147 ± 0.0074
AR-MOEA	0.9917 ± 0.0017	0.9775 ± 0.0009	0.9168 ± 0.0043	0.8832 ± 0.0062	0.9215 ± 0.0035
3DFCH-EMOA	0.9924 ± 0.0011	0.9769 ± 0.0012	0.9182 ± 0.0052	0.8852 ± 0.0043	0.9244 ± 0.0053

Methods \ Datasets	ds6	ds7	ds8	ds9	Average
MP	0.7539 ± 0.0216	0.8643 ± 0.0104	0.8601 ± 0.0096	0.6601 ± 0.0156	0.8370
KP	0.6718 ± 0.0441	0.8168 ± 0.0256	0.8213 ± 0.0256	0.6243 ± 0.0219	0.7967
SpaRSA	0.8183 ± 0.0453	0.8788 ± 0.0594	0.8965 ± 0.0330	0.7181 ± 0.0269	0.8762
OMP	0.7549 ± 0.0221	0.8645 ± 0.0101	0.8615 ± 0.0083	0.6601 ± 0.0156	0.8366
Two_Arch2	0.8060 ± 0.0200	0.9179 ± 0.0038	0.9091 ± 0.0034	0.6993 ± 0.0144	0.8907
NSGA-III	0.8200 ± 0.0144	0.9169 ± 0.0031	0.9112 ± 0.0032	0.7092 ± 0.0134	0.8941
MOEA/DD	0.8155 ± 0.0134	0.9175 ± 0.0018	0.9088 ± 0.0040	0.6989 ± 0.0117	0.8920
RVEA	0.8002 ± 0.0119	0.9156 ± 0.0033	0.9072 ± 0.0029	0.6881 ± 0.0120	0.8875
AR-MOEA	0.8091 ± 0.0156	0.9170 ± 0.0029	0.9087 ± 0.0027	0.7002 ± 0.0114	0.8917
3DFCH-EMOA	0.8149 ± 0.0126	0.9191 ± 0.0032	0.9110 ± 0.0023	0.6991 ± 0.0143	0.8935

and *F-measure* are compared in the following.

Table 6: Mean and standard deviation of *Precision* of ensemble methods on MNIST datasets

Methods \ Datasets	ds1	ds2	ds3	ds4	ds5
MP	0.9902 ± 0.0026	0.9680 ± 0.0071	0.8805 ± 0.0219	0.9106 ± 0.0161	0.8917 ± 0.0176
KP	0.9848 ± 0.0069	0.9604 ± 0.0084	0.8620 ± 0.0202	0.8590 ± 0.0358	0.8707 ± 0.0275
SpaRSA	0.9928 ± 0.0056	0.9823 ± 0.0143	0.9334 ± 0.0482	0.9270 ± 0.0662	0.9520 ± 0.0500
OMP	0.9904 ± 0.0025	0.9679 ± 0.0093	0.8780 ± 0.0198	0.9106 ± 0.0161	0.8903 ± 0.0192
Two_Arch2	0.9949 ± 0.0010	0.9950 ± 0.0010	0.9649 ± 0.0048	0.9915 ± 0.0018	0.9840 ± 0.0023
NSGA-III	0.9951 ± 0.0010	0.9942 ± 0.0016	0.9631 ± 0.0049	0.9891 ± 0.0027	0.9846 ± 0.0023
MOEA/DD	0.9950 ± 0.0016	0.9942 ± 0.0013	0.9617 ± 0.0044	0.9911 ± 0.0020	0.9830 ± 0.0033
RVEA	0.9947 ± 0.0010	0.9941 ± 0.0017	0.9639 ± 0.0052	0.9919 ± 0.0021	0.9833 ± 0.0028
AR-MOEA	0.9954 ± 0.0010	0.9950 ± 0.0008	0.9639 ± 0.0043	0.9904 ± 0.0018	0.9838 ± 0.0032
3DFCH-EMOA	0.9952 ± 0.0009	0.9952 ± 0.0012	0.9656 ± 0.0047	0.9917 ± 0.0019	0.9849 ± 0.0023

Methods \ Datasets	ds6	ds7	ds8	ds9	Average
MP	0.8586 ± 0.0243	0.9026 ± 0.0142	0.9174 ± 0.0117	0.8113 ± 0.0276	0.9034
KP	0.8340 ± 0.0338	0.8823 ± 0.0169	0.8888 ± 0.0249	0.7744 ± 0.0393	0.8796
SpaRSA	0.9457 ± 0.0678	0.9306 ± 0.0528	0.9552 ± 0.0376	0.9343 ± 0.0684	0.9504
OMP	0.8628 ± 0.0255	0.9028 ± 0.0138	0.9171 ± 0.0118	0.8113 ± 0.0276	0.9035
Two_Arch2	0.9804 ± 0.0044	0.9730 ± 0.0042	0.9812 ± 0.0023	0.9773 ± 0.0038	0.9825
NSGA-III	0.9806 ± 0.0031	0.9731 ± 0.0025	0.9795 ± 0.0026	0.9725 ± 0.0064	0.9813
MOEA/DD	0.9796 ± 0.0053	0.9709 ± 0.0035	0.9803 ± 0.0032	0.9747 ± 0.0040	0.9812
RVEA	0.9806 ± 0.0037	0.9720 ± 0.0040	0.9805 ± 0.0021	0.9764 ± 0.0063	0.9819
AR-MOEA	0.9787 ± 0.0041	0.9726 ± 0.0043	0.9794 ± 0.0019	0.9740 ± 0.0056	0.9815
3DFCH-EMOA	0.9826 ± 0.0039	0.9740 ± 0.0023	0.9816 ± 0.0022	0.9769 ± 0.0056	0.9831

The comparison of *Precision* is shown in Table 6. From the table we can see that: 1) The new proposed MOSEL can obtain a higher value of *Precision* than CS and pruning ensemble methods; 2) 3DFCH-EMOA algorithm performs the best on most of the compared datasets. 3) 3DFCH-EMOA and NSGA-III obtain

the best and the second best result on the average *Precision*, respectively. 4) SpaRSA performs better than other CS and pruning algorithms.

Table 7: Mean and standard deviation of *Recall* of ensemble methods on MNIST datasets

Methods \ Datasets	ds1	ds2	ds3	ds4	ds5
MP	0.9859 ± 0.0054	0.9674 ± 0.0068	0.8479 ± 0.0149	0.7874 ± 0.0195	0.8371 ± 0.0277
KP	0.9787 ± 0.0052	0.9551 ± 0.0094	0.7980 ± 0.0296	0.7395 ± 0.0212	0.8153 ± 0.0212
SpaRSA	0.9902 ± 0.0061	0.9722 ± 0.0090	0.8910 ± 0.0363	0.8198 ± 0.0321	0.8858 ± 0.0356
OMP	0.9849 ± 0.0066	0.9674 ± 0.0080	0.8471 ± 0.0149	0.7873 ± 0.0194	0.8326 ± 0.0232
Two_Arch2	0.9960 ± 0.0011	0.9747 ± 0.0018	0.9101 ± 0.0056	0.8278 ± 0.0069	0.8832 ± 0.0102
NSGA-III	0.9945 ± 0.0010	0.9761 ± 0.0009	0.9085 ± 0.0073	0.8352 ± 0.0049	0.8896 ± 0.0079
MOEA/DD	0.9948 ± 0.0023	0.9761 ± 0.0020	0.9108 ± 0.0080	0.8300 ± 0.0082	0.8900 ± 0.0069
RVEA	0.9965 ± 0.0014	0.9743 ± 0.0014	0.9075 ± 0.0061	0.8265 ± 0.0084	0.8782 ± 0.0112
AR-MOEA	0.9957 ± 0.0013	0.9763 ± 0.0011	0.9102 ± 0.0061	0.8325 ± 0.0080	0.8884 ± 0.0070
3DFCH-EMOA	0.9968 ± 0.0010	0.9754 ± 0.0009	0.9106 ± 0.0056	0.8343 ± 0.0061	0.8918 ± 0.0084
Methods \ Datasets	ds6	ds7	ds8	ds9	Average
MP	0.7204 ± 0.0328	0.8597 ± 0.0104	0.8375 ± 0.0138	0.5984 ± 0.0225	0.8269
KP	0.6185 ± 0.0636	0.7989 ± 0.0380	0.7985 ± 0.0309	0.5692 ± 0.0262	0.7857
SpaRSA	0.7555 ± 0.0260	0.8585 ± 0.0526	0.8652 ± 0.0235	0.6143 ± 0.0195	0.8503
OMP	0.7189 ± 0.0324	0.8597 ± 0.0104	0.8400 ± 0.0116	0.5984 ± 0.0225	0.8263
Two_Arch2	0.7158 ± 0.0290	0.8860 ± 0.0068	0.8642 ± 0.0053	0.5714 ± 0.0182	0.8477
NSGA-III	0.7348 ± 0.0208	0.8843 ± 0.0052	0.8689 ± 0.0058	0.5851 ± 0.0173	0.8530
MOEA/DD	0.7291 ± 0.0201	0.8873 ± 0.0052	0.8644 ± 0.0058	0.5717 ± 0.0148	0.8505
RVEA	0.7075 ± 0.0176	0.8832 ± 0.0058	0.8617 ± 0.0050	0.5583 ± 0.0158	0.8437
AR-MOEA	0.7208 ± 0.0221	0.8849 ± 0.0069	0.8649 ± 0.0035	0.5736 ± 0.0140	0.8497
3DFCH-EMOA	0.7265 ± 0.0185	0.8873 ± 0.0054	0.8668 ± 0.0041	0.5712 ± 0.0185	0.8512

The statistical results of *Recall* are listed in Table 7. From the table, we can see that EMOAs methods (i.e., the new proposed MOSEL model) outperform other compared methods on *Recall*. Since in the most of MNIST datasets that we used in this paper, there are far more negative instances than positive samples, the increasing of *Recall* can largely decrease the number of misclassified samples. When comparing results for the *Recall* metric, we can also conclude that the proposed MOSEL methods have clear advantages in the MNIST datasets. While comparing the metric of *Recall*, NSGA-III can obtain the highest value of the average of *Recall* on all the compared datasets, and 3DFCH-EMOA performs better than other methods except NSGA-III.

The *F-measure* is compared in Table 8. From the table we can make some conclusions: 1) The proposed MOSEL outperforms other methods on most of the datasets; 2) NSGA-III and 3DFCH-EMOA can obtain better results than other MOSEL methods; 3) NSGA-III can obtain the best result on the average *F-measure* of all these datasets.

Table 9 shows the mean value and standard deviation of non-zero classifiers of the ensemble. By comparing the results we can conclude that KP and OMP have good performance on sparsity. However, they

Table 8: Mean and standard deviation of *F-measure* of ensemble methods on MNIST datasets

Methods \ Datasets	ds1	ds2	ds3	ds4	ds5
MP	0.9881 ± 0.0030	0.9677 ± 0.0032	0.8636 ± 0.0096	0.8444 ± 0.0132	0.8631 ± 0.0125
KP	0.9817 ± 0.0041	0.9577 ± 0.0059	0.8282 ± 0.0122	0.7943 ± 0.0186	0.8417 ± 0.0154
SpaRSA	0.9915 ± 0.0058	0.9772 ± 0.0113	0.9116 ± 0.0418	0.8699 ± 0.0471	0.9175 ± 0.0413
OMP	0.9876 ± 0.0035	0.9676 ± 0.0032	0.8620 ± 0.0081	0.8443 ± 0.0131	0.8602 ± 0.0116
Two_Arch2	0.9954 ± 0.0006	0.9848 ± 0.0010	0.9367 ± 0.0034	0.9023 ± 0.0041	0.9308 ± 0.0056
NSGA-III	0.9948 ± 0.0008	0.9851 ± 0.0007	0.9350 ± 0.0047	0.9056 ± 0.0031	0.9347 ± 0.0040
MOEA/DD	0.9949 ± 0.0013	0.9851 ± 0.0013	0.9356 ± 0.0049	0.9034 ± 0.0052	0.9342 ± 0.0036
RVEA	0.9956 ± 0.0009	0.9841 ± 0.0010	0.9348 ± 0.0036	0.9017 ± 0.0046	0.9278 ± 0.0063
AR-MOEA	0.9956 ± 0.0009	0.9856 ± 0.0006	0.9363 ± 0.0033	0.9046 ± 0.0052	0.9336 ± 0.0030
3DFCH-EMOA	0.9960 ± 0.0006	0.9852 ± 0.0008	0.9373 ± 0.0040	0.9062 ± 0.0036	0.9360 ± 0.0046

Methods \ Datasets	ds6	ds7	ds8	ds9	Average
MP	0.7829 ± 0.0195	0.8805 ± 0.0091	0.8756 ± 0.0086	0.6883 ± 0.0146	0.8616
KP	0.7076 ± 0.0422	0.8381 ± 0.0229	0.8409 ± 0.0230	0.6553 ± 0.0202	0.8273
SpaRSA	0.8393 ± 0.0394	0.8931 ± 0.0524	0.9079 ± 0.0293	0.7400 ± 0.0239	0.8942
OMP	0.7837 ± 0.0198	0.8807 ± 0.0089	0.8768 ± 0.0074	0.6882 ± 0.0146	0.8612
Two_Arch2	0.8271 ± 0.0184	0.9275 ± 0.0034	0.9190 ± 0.0030	0.7209 ± 0.0138	0.9049
NSGA-III	0.8400 ± 0.0132	0.9266 ± 0.0028	0.9209 ± 0.0029	0.7304 ± 0.0129	0.9081
MOEA/DD	0.8358 ± 0.0123	0.9272 ± 0.0016	0.9187 ± 0.0036	0.7206 ± 0.0113	0.9062
RVEA	0.8218 ± 0.0110	0.9255 ± 0.0029	0.9172 ± 0.0026	0.7102 ± 0.0116	0.9021
AR-MOEA	0.8300 ± 0.0143	0.9267 ± 0.0026	0.9186 ± 0.0024	0.7219 ± 0.0109	0.9059
3DFCH-EMOA	0.8353 ± 0.0115	0.9286 ± 0.0029	0.9206 ± 0.0021	0.7207 ± 0.0138	0.9073

Table 9: Mean and standard deviation of non-zero ensemble weight for each method on MNIST datasets

Methods \ Datasets	ds1	ds2	ds3	ds4	ds5
MP	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00
KP	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
SpaRSA	19.70 ± 16.79	11.80 ± 11.30	20.20 ± 15.99	14.90 ± 14.95	24.40 ± 17.25
OMP	1.00 ± 0.00	1.00 ± 0.00	2.00 ± 0.00	1.60 ± 0.52	1.00 ± 0.00
Two_Arch2	41.30 ± 5.68	42.30 ± 7.67	44.20 ± 4.83	41.50 ± 3.78	39.20 ± 5.22
NSGA-III	15.10 ± 3.41	19.30 ± 2.79	27.10 ± 3.90	25.70 ± 4.83	28.40 ± 2.41
MOEA/DD	13.20 ± 3.94	25.00 ± 5.58	34.50 ± 10.06	50.70 ± 15.56	33.30 ± 8.65
RVEA	67.40 ± 11.35	53.30 ± 10.98	45.20 ± 5.14	49.70 ± 8.12	48.70 ± 5.50
AR-MOEA	24.90 ± 4.48	24.00 ± 3.02	31.70 ± 4.81	30.50 ± 3.24	34.30 ± 4.92
3DFCH-EMOA	83.70 ± 20.23	58.50 ± 25.52	72.40 ± 24.09	64.10 ± 18.88	45.20 ± 8.61

Methods \ Datasets	ds6	ds7	ds8	ds9	Average
MP	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00
KP	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00
SpaRSA	26.30 ± 15.30	21.30 ± 16.60	19.80 ± 16.42	28.30 ± 16.81	20.74
OMP	2.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	2.00 ± 0.00	1.40
Two_Arch2	41.70 ± 6.72	41.50 ± 2.76	40.80 ± 2.86	38.80 ± 4.69	41.26
NSGA-III	27.00 ± 3.50	26.80 ± 3.79	27.30 ± 4.62	28.10 ± 3.28	24.98
MOEA/DD	34.00 ± 4.74	45.00 ± 11.85	41.00 ± 16.25	37.40 ± 9.36	34.90
RVEA	47.60 ± 6.22	50.40 ± 3.66	52.40 ± 7.32	49.20 ± 8.13	51.54
AR-MOEA	34.10 ± 3.81	31.70 ± 2.63	31.60 ± 4.95	32.80 ± 4.21	30.62
3DFCH-EMOA	45.70 ± 7.66	58.50 ± 25.10	48.60 ± 12.28	50.00 ± 9.09	58.52

perform poorly on other metrics. If all values in the table are considered, we can conclude that KP has the best sparseness performance. However, the classification accuracy values of OMP and KP are lower than those of MOSEL methods, as these two algorithms do not find good solutions that balance the performance between classification accuracy and ensemble sparsity. As the performance of sparsity and classification performance are conflicting with each other, a good ensemble method should find the best trade-offs

between them. From the performed experiments we demonstrate that EMOAs are suitable optimization techniques to tackle sparse ensemble problems.

Table 10: Wilcoxon sum-rank test on MNIST datasets: each $x-y-z$ in following table means 3DFCH-EMOA wins x times, losses y times, draws z times

	MP	KP	SpaRSA	OMP	NSGA-III	Two_Arch2	MOEA/DD	RVEA	AR-MOEA
Accuracy	9-0-0	9-0-0	5-1-3	9-0-0	2-0-7	1-0-8	1-0-8	7-0-2	1-0-8
Kappa	9-0-0	9-0-0	5-1-3	9-0-0	2-0-7	1-0-8	1-0-8	6-0-3	1-0-8
Precision	9-0-0	9-0-0	7-0-2	9-0-0	0-0-9	1-0-8	1-0-8	2-0-7	1-0-8
Recall	7-1-1	8-0-1	1-3-5	7-1-1	0-0-9	1-0-8	1-0-8	4-0-5	0-0-9
F-measure	9-0-0	9-0-0	1-1-7	9-0-0	2-0-7	1-0-8	1-0-8	6-0-3	1-0-8
non-zero	0-9-0	0-9-0	0-9-0	0-9-0	0-4-5	0-9-0	0-6-3	0-3-6	0-9-0

As 3DFCH-EMOA has good performance on most of these datasets, a more comprehensive comparison between 3DFCH-EMOA and other ensemble methods is presented in Table 10, which shows the corresponding Wilcoxon sum-rank test [36] results. By comparing the results we can find that 3DFCH-EMOA outperforms CS and pruning ensemble methods significantly on most of the metrics except the non-zero metric on most of the datasets.

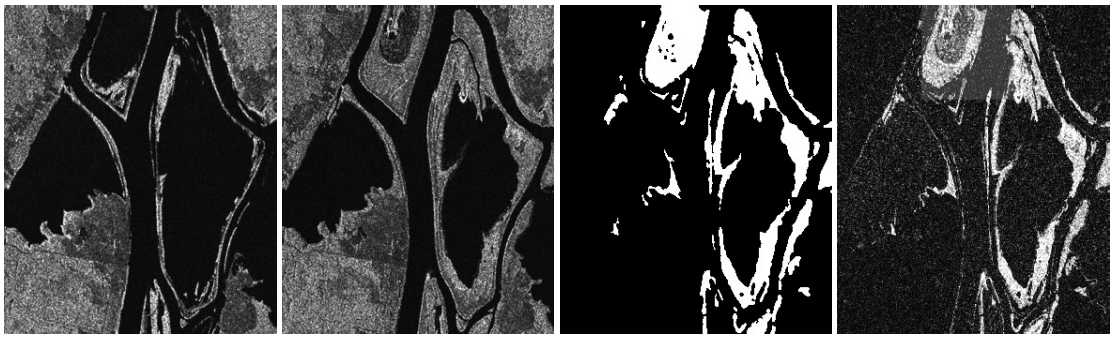
3.5. Experimental results of image change detection

Remote sensing image change detection is a real-world problem that aims to find out the change information that has occurred between two images of the same area taken at different times [42]. It has been applied in many areas, including disaster monitoring, changed target detection and supervision of country resources [43]. Supervised methods have been widely used for remote sensing image change detection [44], as a small amount of labelled data can be used for model training and then the built model can be applied for large-scale image change detection. The change detection problem is an unbalanced classification problem as the proportion of the change area when compared to the total observed area is small. In this part, both synthetic aperture radar (SAR) [45] and optical images are used for the proposed methods evaluation.

3.5.1. Datasets description

Six pairs of remote sensing images are used for classification performance evaluation, details are described in the following. The first dataset is the Ottawa dataset of two SAR images with a spatial resolution of 10m $\times$ 10m and a spatial size of 290 $\times$ 350, acquired in July and August 1997, respectively. They were acquired over the city of Ottawa by the Radarsar SAR sensor and were provided by the Defence Research and Development Canada (DRDC)-Ottawa. Fig. 4(a) and (b) present the flood-afflicted areas and Fig. 4(c)

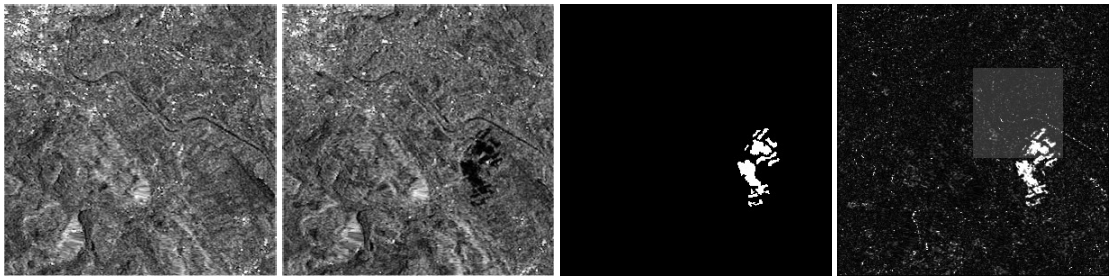
shows the manually defined reference map. The sample patch for model training and validation is marked in blue with a spatial size 100×100 in the log ratio difference image, as shown in Fig. 4(d).



(a) (b) (c) (d)

Figure 4: Multitemporal images relating to Ottawa. (a) Image acquired in July 1997, during the summer flooding, (b) image acquired in August 1997, after the summer flooding, (c) ground truth, (d) initial difference image obtained via the log ratio operator and examples marked in blue extracted for model training and validation.

325 The second dataset is the Bern dataset of two SAR images with a spatial resolution of 10m $\times$ 10m and a spatial size of 301×301. They were acquired over the city of Bern, Switzerland by the European Remote Sensing 2 satellite SAR sensor in April and May 1999, respectively. Fig. 5 shows the two images, manually defined reference map and training image patch with a spatial size 100×100 in the log ratio difference image.

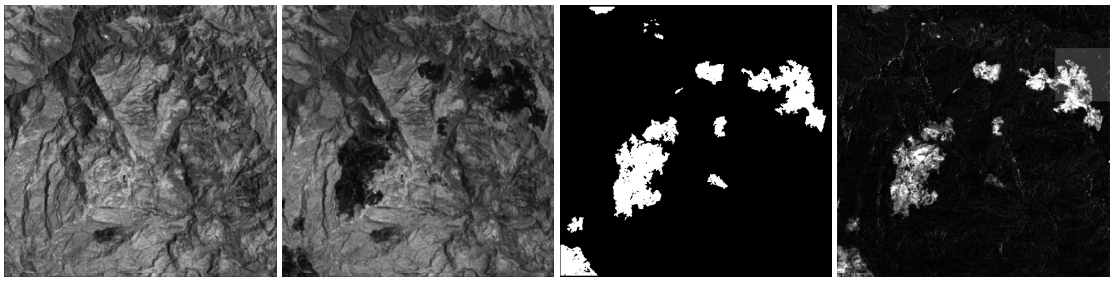


(a) (b) (c) (d)

Figure 5: Multitemporal images relating to the city of Bern. (a) Image acquired in April 1999, (b) image acquired in May 1999, (c) ground truth and (d) examples extracted for model training and validation.

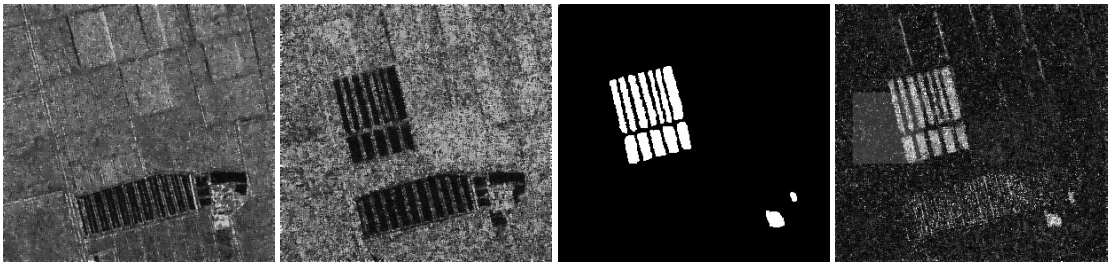
330 The third dataset is the Mexico dataset of two optical images acquired by Landsat-7 (US satellite) in April 2000 and May 2002, respectively. These two images are extracted from Band 4 of the ETM+ images. The sizes of both images are 512×512 pixels. This dataset shows the vegetation damage after the forest

fire in urban Mexico. Fig. 6(a)-(d) show the two images, reference map and example patch with a spatial size 100×100, respectively.



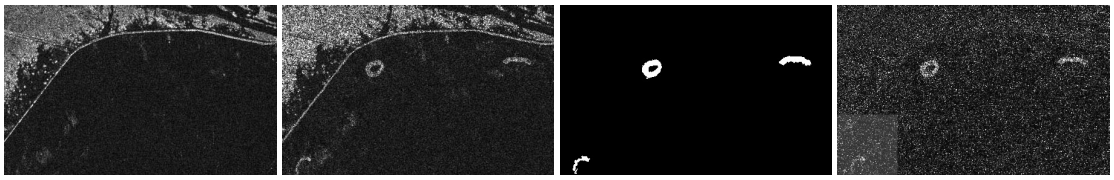
(a) (b) (c) (d)

Figure 6: Multitemporal images relating to the city of Mexico. (a) Optical image acquired in 2000, (b) optical image acquired in 2002, (c) ground truth and (d) examples extracted for model training and validation.



(a) (b) (c) (d)

Figure 7: Multitemporal images relating to Farmland of Yellow River Estuary. (a) SAR image acquired in June 2008, (b) SAR image acquired in June 2009, (c) ground truth and (d) examples extracted for model training and validation.



(a) (b) (c) (d)

Figure 8: Multitemporal images relating to Coastline of Yellow River Estuary. (a) SAR image acquired in June 2008, (b) SAR image acquired in June 2009, (c) ground truth and (d) examples extracted for model training and validation.

The 4-6th datasets are the selected from the Yellow River in eastern China of two SAR images captured by Radarsat-2 (Canadian satellite) with a spatial resolution 8m $\times$ 8m in July 2008 and June 2009, respectively. Note that the two SAR images are single-look and four-look, respectively, which increases the

(a) (b) (c) (d)

Figure 9: Multitemporal images relating to Inland water of Yellow River Estuary. (a) SAR image acquired in June 2008, (b) SAR image acquired in June 2009, (c) ground truth and (d) examples extracted for model training and validation.

difficulty of change detection. These datasets include different typical areas, including farmlands, coastline and inland water. Fig. 7 shows the changed areas that appear as newly reclaimed farmlands, with a spatial size 306×291. Fig. 8 shows the coastline where the changed areas are relatively small, with a spatial size 450×280. Inland water where the changed areas are concentrated on the borderline of the river is shown in Fig. 9. The spatial size of Inland water is 291×444.

Table 11: The details of remote sensing datasets

Datasets / Size	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water
Image spatial size	290×350	301×301	512×512	306×291	450×280	291×444
Sample patch size	100×100	100×100	100×100	80×80	80×80	100×100

The spatial and sample patch sizes of these remote sensing dataset are listed in Table 11. In this part, discrete wavelet transform [46], gray-level co-occurrence matrix (CLCM) [47] and Gabor filter bank [6] are selected to extract features for each pixel of log difference images. The dimension of the feature is 38, i.e., each pixel of the log difference image is represented by a 38 dimension vector. For each dataset, 2/3 samples from the training patch are randomly selected for model training and the remaining 1/3 samples are selected for validation. The whole log difference images are used for testing.

3.5.2. Experimental results and discussion

The mean and standard deviation of the classification accuracy are shown in Table 12. By comparing all the results we can conclude that: 1) OMP performs the best on Ottawa and Farmland datasets; 2) The methods of MOSEL outperform CS and pruning ensemble methods on most of the datasets except Ottawa and Farmland; 3) 3DFCH-EMOA can obtain the highest accuracy except for the Farmland dataset.

Table 12: Mean and standard deviation of *Accuracy* of ensemble methods on change detection datasets

Methods \ Datasets	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water	Average
MP	0.9190±0.0069	0.9880±0.0020	0.9665±0.0043	0.9145±0.0149	0.9855±0.0062	0.9693±0.0027	0.9571
KP	0.8882±0.0657	0.9898±0.0014	0.9670±0.0059	0.9204±0.0154	0.9866±0.0054	0.9659±0.0078	0.9530
SpaRSA	0.9123±0.0235	0.9909±0.0031	0.9636±0.0071	0.9186±0.0108	0.9831±0.0086	0.9677±0.0079	0.9560
OMP	0.9276±0.0023	0.9887±0.0018	0.9682±0.0038	0.9239±0.0050	0.9855±0.0061	0.9694±0.0029	0.9605
Two_Arch2	0.9263±0.0027	0.9931±0.0004	0.9725±0.0015	0.9205±0.0020	0.9908±0.0006	0.9731±0.0009	0.9627
NSGA-III	0.9267±0.0030	0.9930±0.0005	0.9723±0.0014	0.9227±0.0034	0.9907±0.0008	0.9735±0.0010	0.9632
MOEA/DD	0.9275±0.0018	0.9932±0.0004	0.9723±0.0014	0.9229±0.0030	0.9911±0.0006	0.9739±0.0010	0.9635
RVEA	0.9262±0.0012	0.9929±0.0004	0.9722±0.0013	0.9206±0.0028	0.9909±0.0006	0.9735±0.0009	0.9627
AR-MOEA	0.9268±0.0025	0.9929±0.0004	0.9716±0.0017	0.9214±0.0040	0.9906±0.0003	0.9734±0.0012	0.9628
3DFCH-EMOA	0.9276±0.0025	0.9934±0.0004	0.9729±0.0014	0.9227±0.0013	0.9911±0.0005	0.9741±0.0007	0.9637

The statistical results of *Kappa* are shown in Table 13. By comparing the results on the table, we can conclude that: 1) OMP outperforms other CS and pruning ensemble methods; 2) 3DFCH-EMOA and MOEA/DD perform better than other MOSEL methods on most of the datasets; 3) 3DFCH-EMOA can obtain the best result on the average *Kappa* of the six remote sensing datasets.

Table 13: Mean and standard deviation of *Kappa* of ensemble methods on change detection datasets

Methods \ Datasets	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water	Average
MP	0.6782±0.0224	0.5702±0.0442	0.7970±0.0315	0.4747±0.0530	0.3616±0.1256	0.5565±0.0216	0.5730
KP	0.6180±0.1322	0.5811±0.0584	0.8042±0.0383	0.4867±0.0651	0.3327±0.2237	0.5291±0.0736	0.5586
SpaRSA	0.6683±0.0700	0.6425±0.0694	0.7776±0.0448	0.5013±0.0498	0.3540±0.1189	0.5578±0.0632	0.5836
OMP	0.7170±0.0077	0.5858±0.0475	0.8072±0.0278	0.5239±0.0176	0.3464±0.0932	0.5620±0.0326	0.5904
Two_Arch2	0.7177±0.0078	0.6798±0.0386	0.8376±0.0105	0.5204±0.0088	0.3463±0.1073	0.5970±0.0111	0.6165
NSGA-III	0.7179±0.0080	0.6885±0.0335	0.8360±0.0098	0.5273±0.0124	0.3679±0.1192	0.6017±0.0118	0.6232
MOEA/DD	0.7209±0.0054	0.6913±0.0330	0.8364±0.0097	0.5310±0.0131	0.3887±0.0859	0.6023±0.0101	0.6284
RVEA	0.7186±0.0035	0.6725±0.0298	0.8354±0.0091	0.5234±0.0107	0.3511±0.1072	0.5985±0.0083	0.6166
AR-MOEA	0.7187±0.0077	0.6814±0.0335	0.8309±0.0118	0.5222±0.0159	0.3322±0.0518	0.5981±0.0132	0.6139
3DFCH-EMOA	0.7211±0.0064	0.7020±0.0271	0.8407±0.0095	0.5279±0.0072	0.3873±0.0947	0.6064±0.0092	0.6309

The metrics of *Precision* and *Recall* are compared in Table 14 and Table 15, respectively. By comparing the results we can find out that: 1) The new proposed MOSEL methods outperform other methods on most of the datasets; 2) 3DFCH-EMOA obtained the best result and MOEA/DD obtained the second best result on the average *Precision*; 3) 3DFCH-EMOA obtained the best result and KP obtained the second best result on the average *Recall*; 4) The two metrics, i.e., *Precision* and *Recall* are conflicting with each

Table 14: Mean and standard deviation of *Precision* of ensemble methods on change detection datasets

Methods \ Datasets	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water	Average
MP	0.7846±0.0436	0.5336±0.0727	0.8790±0.0128	0.3915±0.0467	0.3810±0.1346	0.6030±0.0502	0.5954
KP	0.6779±0.1477	0.6124±0.0675	0.8639±0.0271	0.4119±0.0658	0.3597±0.1247	0.5655±0.1097	0.5819
SpaRSA	0.7412±0.0904	0.6834±0.1484	0.8720±0.0435	0.4074±0.0396	0.4298±0.2348	0.5873±0.1053	0.6202
OMP	0.8005±0.0143	0.5548±0.0675	0.8896±0.0120	0.4265±0.0191	0.3776±0.1211	0.6021±0.0491	0.6085
Two_Arch2	0.7816±0.0169	0.8152±0.0310	0.8944±0.0038	0.4161±0.0074	0.7171±0.0551	0.6674±0.0188	0.7153
NSGA-III	0.7855±0.0208	0.7865±0.0353	0.8958±0.0038	0.4237±0.0125	0.6647±0.0532	0.6757±0.0234	0.7053
MOEA/DD	0.7886±0.0129	0.8052±0.0309	0.8919±0.0031	0.4251±0.0113	0.7310±0.0339	0.6875±0.0288	0.7216
RVEA	0.7768±0.0095	0.8151±0.0318	0.8922±0.0024	0.4171±0.0097	0.7455±0.0644	0.6782±0.0213	0.7208
AR-MOEA	0.7852±0.0182	0.7914±0.0307	0.8951±0.0027	0.4191±0.0146	0.7009±0.0700	0.6742±0.0262	0.7110
3DFCH-EMOA	0.7898±0.0186	0.8177±0.0235	0.8930±0.0025	0.4237±0.0053	0.7223±0.0275	0.6895±0.0160	0.7227

other. Generally, a method which has a good performance on one objective will not necessarily have a good performance on another objective. 3DFCH-EMOA can find a good trade-off between these two metrics.

Table 15: Mean and standard deviation of *Recall* of ensemble methods on change detection datasets

Methods \ Datasets	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water	Average
MP	0.6763±0.0268	0.6358±0.0577	0.7617±0.0512	0.7619±0.0442	0.3883±0.1587	0.5479±0.0284	0.6287
KP	0.7069±0.0341	0.5750±0.1011	0.7861±0.0539	0.7354±0.0508	0.5228±0.2162	0.5440±0.0831	0.6450
SpaRSA	0.7038±0.0309	0.6352±0.0564	0.7364±0.0533	0.8050±0.0543	0.4424±0.2395	0.5709±0.0315	0.6489
OMP	0.7227±0.0090	0.6404±0.0620	0.7696±0.0450	0.8206±0.0186	0.3553±0.1034	0.5585±0.0418	0.6445
Two_Arch2	0.7422±0.0115	0.5927±0.0667	0.8150±0.0203	0.8493±0.0102	0.2415±0.0954	0.5634±0.0117	0.6340
NSGA-III	0.7389±0.0131	0.6219±0.0605	0.8110±0.0189	0.8466±0.0088	0.2700±0.1220	0.5654±0.0174	0.6423
MOEA/DD	0.7405±0.0107	0.6149±0.0613	0.8151±0.0171	0.8563±0.0123	0.2740±0.0789	0.5584±0.0183	0.6432
RVEA	0.7484±0.0092	0.5804±0.0522	0.8131±0.0155	0.8588±0.0151	0.2438±0.1039	0.5582±0.0097	0.6338
AR-MOEA	0.7405±0.0154	0.6074±0.0565	0.8032±0.0200	0.8442±0.0088	0.2245±0.0492	0.5606±0.0144	0.6300
3DFCH-EMOA	0.7397±0.0115	0.6216±0.0431	0.8214±0.0173	0.8486±0.0117	0.2763±0.0923	0.5631±0.0146	0.6451

The statistical results of *F-measure* is listed in Table 16. *F-measure* is a comprehensive consideration of *Precision* and *Recall*. By comparing the results we can make conclusions: 1) The proposed MOSEL method performs better than other methods on most of the remote sensing datasets; 2) 3DFCH-EMOA outperforms others on most of the datasets; 3) MOEA/DD performs the second best on the average *F-measure*.

Table 16: Mean and standard deviation of *F-measure* of ensemble methods on change detection datasets

Methods \ Datasets	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water	Average
MP	0.7254±0.0184	0.5762±0.0434	0.8153±0.0293	0.5160±0.0475	0.3684±0.1241	0.5724±0.0204	0.5956
KP	0.6829±0.0974	0.5862±0.0580	0.8223±0.0352	0.5259±0.0587	0.4015±0.1448	0.5465±0.0706	0.5942
SpaRSA	0.7200±0.0557	0.6470±0.0679	0.7975±0.0411	0.5405±0.0453	0.3611±0.1174	0.5744±0.0594	0.6068
OMP	0.7595±0.0063	0.5914±0.0467	0.8245±0.0259	0.5610±0.0157	0.3533±0.0918	0.5778±0.0313	0.6112
Two_Arch2	0.7612±0.0063	0.6832±0.0385	0.8527±0.0098	0.5585±0.0079	0.3496±0.1080	0.6108±0.0107	0.6360
NSGA-III	0.7612±0.0062	0.6920±0.0333	0.8512±0.0091	0.5646±0.0111	0.3715±0.1194	0.6152±0.0114	0.6426
MOEA/DD	0.7636±0.0045	0.6947±0.0328	0.8517±0.0089	0.5681±0.0118	0.3922±0.0863	0.6156±0.0097	0.6477
RVEA	0.7623±0.0030	0.6759±0.0297	0.8507±0.0084	0.5614±0.0097	0.3543±0.1078	0.6121±0.0078	0.6361
AR-MOEA	0.7619±0.0064	0.6849±0.0334	0.8465±0.0109	0.5600±0.0143	0.3357±0.0522	0.6118±0.0127	0.6335
3DFCH-EMOA	0.7637±0.0049	0.7053±0.0270	0.8556±0.0088	0.5652±0.0066	0.3908±0.0951	0.6197±0.0089	0.6500

Table 17 shows the mean value and standard deviation of non-zero classifiers of the ensemble weight.

Table 17: Mean and standard deviation of non-zero ensemble weight for each method on change detection datasets

Methods \ Datasets	Ottawa	Bern	Mexico	Farmland	Coastline	Inland water	Average
MP	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00 ± 0.00	26.00
KP	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00
SpaRSA	19.10 ± 16.30	23.90 ± 17.11	14.60 ± 15.89	28.50 ± 15.79	12.10 ± 15.20	17.80 ± 16.84	19.33
OMP	21.80 ± 2.62	1.00 ± 0.00	2.70 ± 0.48	15.20 ± 9.37	1.00 ± 0.00	1.00 ± 0.00	7.12
Two_Arch2	49.80 ± 6.16	48.00 ± 5.37	45.90 ± 4.46	50.80 ± 4.96	49.00 ± 7.12	49.10 ± 4.36	48.77
NSGA-III	37.60 ± 3.98	33.40 ± 4.55	35.50 ± 3.14	38.00 ± 5.64	27.50 ± 4.84	34.90 ± 5.45	34.48
MOEA/DD	46.60 ± 11.94	53.20 ± 13.60	50.40 ± 17.69	42.30 ± 5.03	53.20 ± 11.59	42.30 ± 11.66	48.00
RVEA	49.90 ± 8.09	60.20 ± 5.92	55.00 ± 7.59	45.70 ± 4.81	67.70 ± 7.26	56.40 ± 6.24	55.82
AR-MOEA	43.90 ± 4.77	39.60 ± 5.25	41.90 ± 5.78	45.70 ± 4.47	37.60 ± 3.57	41.50 ± 5.52	41.70
3DFCH-EMOA	71.40 ± 17.49	62.30 ± 10.63	85.60 ± 21.82	70.20 ± 16.25	63.00 ± 24.81	74.40 ± 23.98	71.15

By comparing the results we can conclude that KP and OMP have good performance on sparsity. However, they perform poorly on *Accuracy* and *Kappa* metrics.

Table 18: Wilcoxon sum-rank test on change detection datasets: each $x - y - z$ in this table means that 3DFCH-EMOA wins x times, losses y times, draws z times

	MP	KP	SpaRSA	OMP	NSGA-III	Two_Arch2	MOEA/DD	RVEA	AR-MOEA
Accuracy	5-0-1	5-0-1	5-0-1	4-0-2	2-0-4	0-0-6	0-0-6	2-0-4	1-0-5
Kappa	5-0-1	5-0-1	4-0-2	3-0-3	0-0-6	0-0-6	0-0-6	1-0-5	0-0-6
Precision	5-0-1	5-0-1	3-0-3	3-0-3	2-0-4	2-0-4	0-0-6	0-0-6	1-0-5
Recall	3-0-3	2-1-3	3-0-3	3-0-3	0-0-6	0-0-6	0-0-6	0-0-6	1-0-5
F-measure	5-0-1	5-0-1	4-0-2	3-0-3	0-0-6	0-0-6	0-0-6	1-0-5	0-0-6
non-zeros	0-6-0	0-6-0	0-6-0	0-6-0	0-5-1	0-6-0	0-4-2	0-2-4	0-6-0

As 3DFCH-EMOA has good performance on most of the compared metrics, we make a more comprehensive comparison between 3DFCH-EMOA and other ensemble methods. The Wilcoxon sum-rank test results are listed in Table 18. By comparing the results we can find out that 3DFCH-EMOA outperforms CS and pruning ensemble methods significantly on accuracy and *Kappa* metrics for most of the datasets.

4. Conclusions

In this paper, we proposed the multiobjective sparse ensemble learning model and analyzed its properties in the ADET space. Firstly, MOSEL is modeled as ADCH maximization problem, and the relationship between the sparsity and the performance of ensemble classifiers on the ADET space is explained. Secondly, sparse real encoding is designed as a bridge between MOSEL and EMOAs, and six EMOAs were used to find a sparse ensemble classifier with good performance. Thirdly, an adaptive MOSEL classifier selection algorithm was proposed to select the most suitable ensemble classifier for a given dataset. Experimental results based on well-known MNIST and remote sensing change detection datasets show that the proposed MOSEL performs significantly better than conventional ensemble learning methods. However,

the distribution of MOSEL solutions obtained by several EMOAs is not even. To find evenly distributed solutions MOSEL must be studied further. Besides, the generation of candidate classifiers was not taken into consideration and the importance of each dimension of data not studied in depth. In the future, the generation of candidate classifiers can be further studied, which can provide some useful rules related to the corresponding data mining tasks.

Acknowledgment

The authors would like to thank the editor and anonymous reviewers for their very competent comments and suggestions. This work was supported by the Fundamental Research Funds for the Central Universities (No. 2018XKQYMS27).

References

[1] S. Piri, D. Delen, T. Liu, H. M. Zolbanin, A data analytics approach to building a clinical decision support system for diabetic retinopathy: Developing and deploying a model ensemble, Decision Support Systems 101 (2017) 12 – 27.

[2] R. Gupta, K. Audhkhasi, S. Narayanan, Training ensemble of diverse classifiers on feature subsets, in: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on, 2014, pp. 2927–2931.

[3] A. Riccardi, F. Fernandez-Navarro, S. Carloni, Cost-sensitive adaboost algorithm for ordinal regression based on extreme learning machine, IEEE Transactions on Cybernetics 44 (10) (2014) 1898–1909.

[4] Y. Liu, C. Jiang, H. Zhao, Using contextual features and multi-view ensemble learning in product defect identification from online discussion forums, Decision Support Systems.

[5] P. du Jardin, Failure pattern-based ensembles applied to bankruptcy forecasting, Decision Support Systems 107 (2018) 64 – 77.

[6] Z. Zhao, L. Jiao, F. Liu, J. Zhao, Semisupervised discriminant feature learning for SAR image category via sparse ensemble, IEEE Transactions on Geoscience and Remote Sensing 54 (6) (2016) 3532–3547.

[7] L. Breiman, Bagging predictors, Machine Learning 24 (1996) 123–140.

[8] J. H. Friedman, Greedy function approximation: A gradient boosting machine., Annals of Statistics 29 (5) (2001) 1189–1232.

[9] T. K. Ho, The random subspace method for constructing decision forests, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 832–844.

[10] J. Rodriguez, L. Kuncheva, C. Alonso, Rotation forest: A new classifier ensemble method, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (10) (2006) 1619–1630.

[11] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, C. Coello, Survey of multiobjective evolutionary algorithms for data mining: Part II, IEEE Transactions on Evolutionary Computation 18 (1) (2014) 20–35.

[12] M. Asafuddoula, B. Verma, M. Zhang, A divide-and-conquer based ensemble classifier learning by means of many-objective optimization, IEEE Transactions on Evolutionary Computation (2017) 1–1doi:10.1109/TEVC.2017.2782826.

[13] C. Zhang, P. Lim, A. K. Qin, K. C. Tan, Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics, IEEE Transactions on Neural Networks and Learning Systems 28 (10) (2017) 2306–2318.

[14] W. A. Albukhanajer, Y. Jin, J. A. Briffa, Classifier ensembles for image identification using multi-objective pareto features, Neurocomputing 238 (2017) 316 – 327.

[15] Z.-h. Zhou, J. Wu, W. Tang, Ensembling neural networks: Many could be better than all, Artificial Intelligence 137 (2002) 239–263.

[16] H. Chen, P. Tiho, X. Yao, Predictive ensemble pruning by expectation propagation, IEEE Transactions on Knowledge and Data Engineering 21 (7) (2009) 999–1013.

[17] G. Martínez-muñoz, D. Hernández-lobato, A. Suárez, An analysis of ensemble pruning techniques based on ordered aggregation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2009) 245–259.

[18] S. Mao, L. Jiao, L. Xiong, S. Gou, Greedy optimization classifiers ensemble based on diversity, Pattern Recognition 44 (6) (2011) 1245–1261.

[19] L. Zhang, W.-D. Zhou, Sparse ensembles using weighted combination methods based on linear programming, Pattern Recognition 44 (1) (2011) 97 – 106.

[20] L. Li, X. Yao, R. Stolkin, M. Gong, S. He, An evolutionary multiobjective approach to sparse reconstruction, IEEE Transactions on Evolutionary Computation 18 (6) (2014) 827–845.

[21] Y.-L. Chen, C.-L. Chang, C.-S. Yeh, Emotion classification of youtube videos, Decision Support Systems 101 (Supplement C) (2017) 40 – 50.

[22] D. Donoho, Compressed sensing, IEEE Transactions on Information Theory 52 (4) (2006) 1289–1306.

[23] L. Li, R. Stolkin, L. Jiao, F. Liu, S. Wang, A compressed sensing approach for efficient ensemble learning, Pattern Recognition 47 (10) (2014) 3451–3465.

[24] S. Wright, R. Nowak, M. Figueiredo, Sparse reconstruction by separable approximation, IEEE Transactions on Signal Processing 57 (7) (2009) 2479–2493.

[25] G. Davis, S. Mallat, M. Avellaneda, Adaptive greedy approximations, Constructive Approximation 13 (1997) 57–98.

[26] K.-C. Toh, S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems, Pacific Journal of Optimization 6 (3) (2010) 615–640.

[27] M. D. Plumbley, Recovery of sparse representations by polytope faces pursuit, in: International Conference on Independent Component Analysis and Signal Separation, 2006, pp. 206–213.

[28] A. F. Martin, G. R. Doddington, T. Kamm, M. Ordowski, M. A. Przybocki, The DET curve in assessment of decision task performance, in: European Conference on Speech Communication and Technology, Eurospeech 1997, Rhodes, Greece, September, 1997, pp. 1895–1898.

[29] A. Mattiussi, M. Rosano, P. Simeoni, A decision support system for sustainable energy supply combining multi-objective and multi-attribute analysis: An australian case study, Decision Support Systems 57 (Supplement C) (2014) 150 – 159.

[30] H. Wang, L. Jiao, X. Yao, Two_Arch2: An improved two-archive algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 19 (4) (2015) 524–541.

[31] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints, IEEE Transactions on Evolutionary Computation 18 (4) (2014) 577–601.

[32] K. Li, K. Deb, Q. Zhang, S. Kwong, An evolutionary many-objective optimization algorithm based on dominance and decomposition, IEEE Transactions on Evolutionary Computation 19 (5) (2015) 694–716.

[33] R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A reference vector guided evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 773–791.

[34] Y. Tian, R. Cheng, X. Zhang, F. Cheng, Y. Jin, An indicator based multi-objective evolutionary algorithm with reference point adaptation for better versatility, IEEE Transactions on Evolutionary Computation (2017) 1–1doi:10.1109/TEVC.2017.2749619.

[35] J. Zhao, V. Basto Fernandes, L. Jiao, I. Yevseyeva, A. Maulana, R. Li, T. Bäck, K. Tang, M. T.M. Emmerich, Multiobjective optimization of classifiers by means of 3D convex-hull-based evolutionary algorithms, Information Sciences 367–368 (2016) 80–104.

[36] J. Zhao, L. Jiao, F. Liu, V. B. Fernandes, I. Yevseyeva, S. Xia, M. T. Emmerich, 3d fast convex-hull-based evolutionary multiobjective optimization algorithm, Applied Soft Computing 67 (2018) 322 – 336.

[37] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, in: IEEE, 1998, pp. 2278–2324.

[38] P. Wang, M. Emmerich, R. Li, K. Tang, T. Bäck, X. Yao, Convex hull-based multi-objective genetic programming for maximizing receiver operator characteristic performance, IEEE Transactions on Evolutionary Computation 19 (2) (2015) 188–200.

[39] I. Mendialdua, A. Arruti, E. Jauregi, E. Lazkano, B. Sierra, Classifier subset selection to construct multi-classifiers by means of estimation of distribution algorithms, Neurocomputing 157 (2015) 46–60.

[40] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. Mclachlan, A. Ng, B. Liu, P. S. Yu, Top 10 algorithms in data mining, Knowledge and Information Systems 14 (1) (2008) 1–37.

[41] G. H. Rosenfield, A coefficient of agreement as a measure of thematic classification accuracy, Photogrammetric Engineering & Remote Sensing 52 (2) (1986) 223–227.

[42] M. Gong, J. Zhao, J. Liu, Q. Miao, L. Jiao, Change detection in synthetic aperture radar images based on deep neural networks, IEEE Transactions on Neural Networks and Learning Systems 27 (1) (2016) 125–138.

[43] Y. Zheng, L. Jiao, H. Liu, X. Zhang, B. Hou, S. Wang, Unsupervised saliency-guided sar image change detection, Pattern Recognition 61 (2017) 309 – 326.

[44] R. J. Radke, S. Andra, O. Al-Kofahi, B. Roysam, Image change detection algorithms: a systematic survey, IEEE Transactions on Image Processing 14 (3) (2005) 294–307.

[45] J. Liu, M. Gong, K. Qin, P. Zhang, A deep convolutional coupling network for change detection based on heterogeneous

optical and radar images, IEEE Transactions on Neural Networks and Learning Systems 29 (3) (2018) 545–559.

[46] G. Akbarizadeh, A new statistical-based kurtosis wavelet energy feature for texture recognition of sar images, IEEE Transactions on Geoscience & Remote Sensing 50 (11) (2012) 4358–4368.

[47] B. Hou, X. Zhang, N. Li, Mpm sar image segmentation using feature extraction and context model, IEEE Geoscience & Remote Sensing Letters 9 (6) (2012) 1041–1045.