Expert Systems



Article

DOI: 10.1111/j.1468-0394.2012.00637.x

Cluster-based ensemble of classifiers

Ashfaqur Rahman[1] and Brijesh Verma[2]

(1) Intelligent Sensing and Systems Lab, CSIRO, TAS, Australia
Email: ashfaqur.rahman@csiro.au
(2) CINS, CQUni, Rockhampton, QLD, Australia
Email: b.verma@cqu.edu.au

Abstract: *This paper presents cluster-based ensemble classifier – an approach toward generating ensemble of classifiers using multiple clusters within classified data. Clustering is incorporated to partition data set into multiple clusters of highly correlated data that are difficult to separate otherwise and different base classifiers are used to learn class boundaries within the clusters. As the different base classifiers engage on different difficult-to-classify subsets of the data, the learning of the base classifiers is more focussed and accurate. A selection rather than fusion approach achieves the final verdict on patterns of unknown classes. The impact of clustering on the learning parameters and accuracy of a number of learning algorithms including neural network, support vector machine, decision tree and k-NN classifier is investigated. A number of benchmark data sets from the UCI machine learning repository were used to evaluate the cluster-based ensemble classifier and the experimental results demonstrate its superiority over bagging and boosting.*

Keywords: ensemble classifier, cluster-based ensemble, multiple classifier system

1. Introduction

In an "ensemble classifier" system (Tsoumakas *et al.*, 2004; Le *et al.*, 2007; Parikh & Polikar, 2007; Muhlbaier *et al.*, 2009), a number of *base classifiers* are trained on a data set and the class of a new pattern is predicted by fusing the decisions of the base classifiers. Ensemble classifiers are also known as multiple classifier systems, committee of classifiers or mixture of experts. The construction of the base classifiers in the ensemble is constrained by *diversity* that forces the different base classifier errors to be uncorrelated. The base classifiers' decisions are fused to produce the final verdict on the class of a pattern. It is necessary that the individual classifiers are accurate and diverse for an ensemble classifier to be more accurate than its base counterparts.

The base classifiers in the existing ensemble-based approaches (Schapire, 1990; Breiman, 1996; Freund & Schapire, 1997; Todorovski & Dzeroski, 2003; Nanni & Lumini, 2006; Rodriguez & Maudes, 2008; Chen & Kamel, 2009; Munoz *et al.*, 2009; Pedrajas, 2009; Saffari *et al.*, 2009) are made diverse by training them on different subsets of the training data and the subset selection algorithm varies among the approaches. The decision of multiple base classifiers on the same pattern is mapped into a single decision using some fusion method (Kittler *et al.*, 1998; Kuncheva *et al.*, 2001; Polikar, 2006; Wanas *et al.*, 2006; Ko *et al.*, 2007; Cevikalp & Polikar, 2008; Terrades *et al.*, 2009). We aim to address the subset selection issue from a different angle using a clustering approach. *Clustering* refers to the existence of multiple clusters within the classified data. Each cluster normally contains patterns from multiple classes that are close in Euclidian space and difficult to separate. Base classifiers if trained on these non-overlapping clusters, will learn decision boundaries on difficult-to-classify small subsets. The new patterns can be classified by finding the appropriate cluster and using the corresponding base classifier to predict the class. As a pattern can belong to one cluster only, a base classifier selection approach can be used for predicting the class.

With the above motivation the research presented in this paper aims to (1) develop a cluster-based ensemble classifier with atomic/non-atomic cluster identification, multiple base classifier based non-atomic cluster learning with a selection rather than fusion approach, (2) investigate the impact of incorporating clustering on the classification accuracy by conducting a number of experiments on benchmark data sets, (3) analyse the classification accuracies by using k Nearest Neighbour, support vector machine (SVM), decision tree (DT) and neural network (NN) as the base classifiers to learn decision boundaries within the clusters and (4) conduct a comparative analysis with bagging and boosting.

This paper is organized as follows. Some related works are presented in Section 2. Section 3 presents the proposed ensemble approach by detailing the underlying theoretical framework and the methodology. Section 4 describes the experimental setup and the results obtained on different benchmark data sets. Section 5 concludes the paper.

2. Related works

This section presents some algorithms on ensemble classifier generation and fusion. An ensemble generation algorithm called bootstrap aggregating or "Bagging" was proposed by Breiman (1996). Bagging achieves diversity by training the base classifiers on different training data subsets randomly drawn (with replacement) from the entire training set. Bagging is effective when available data is of limited size. Schapire (1990) proposed a method called *boosting* that creates an ensemble of classifiers by re-sampling the training data, however, by providing most informative training data for each consecutive base classifier. Freund and Schapire (1997)

© 2012 Wiley Publishing Ltd

extended the concept to AdaBoost that is a more generalized version of boosting. It trains a classifier on instances that previous classifiers fail to classify.

There are a number of variants of the bagging and boosting ensemble generation methods. Nanni and Lumini (2006) proposed an ensemble generation method called Fuzzy Bagging that uses Fuzzy C-Means to obtain several training set and then train a classifier on each training set and the output of classifiers were fused using majority win. Todorovski and Dzeroski (2003) proposed a Meta DT approach Todorovski & Dzeroski, where the leaves of the DT specify which base classifier should be used to obtain a prediction. Pedraja (2009) used a modified training instance selection method with boosting so that the selected subset of the instances achieves the same performance as the whole training set. Munoz *et al*. (2009) investigated the impact of changing the order of aggregation in Bagging. Rodriguez and Maudes (2008) proposed a variant of AdaBoost where a classifier is formed by the last r selected weak classifiers instead of only the last weak classifier. Among others Saffari *et al*.'s (2009) multi-class semi-supervised boosting and Chen and Kamel's (2009) adaptive generation and aggregation approach are worth mentioning. Space decomposition methods (Kuncheva, 2000; Tang *et al*., 2002; Rokach *et al*., 2003; Xiuping & Richards, 2003; Nasierding *et al*., 2009) partition data sets for efficient learning. These methods were actually designed to reduce the complexity of learning large data sets (Rokach *et al*., 2003).

Alongside ensemble generation methods a number approaches are also developed for fusing the decision of the base classifiers. Commonly used decision fusion methods (Polikar, 2006) include majority voting, weighted majority voting, behaviour knowledge space, and Borda count. A number of algebraic combiners (Kittler *et al*., 1998) including mean rule, weighted average, trimmed mean, min/max/median rule, product rule, and generalized mean are also used to fuse the class confidence measures produced by the base classifiers. Kuncheva *et al*. (2001) proposed a fusion method for class confidence measures using a decision template for producing the final verdict.

Ko *et al*. (2007) proposed a pairwise fusion matrix (PFM) transformation that takes into account the interaction of classifiers in a pairwise manner and transforms crisp class label outputs into class probability outputs so that it becomes suitable for most kinds of existing fusion functions for combining classifiers. Wanas *et al*. (2006) proposed an adaptive fusion method that identifies the recognition strengths and weaknesses of the different base classifiers to guide the fusion process. Cevikalp & Polikar (2008) proposed a fusion approach where local recognition accuracies of the classifiers are estimated near a query sample by utilizing its nearest neighbours, and the best weights of classifiers are found using these estimates using convex quadratic optimization. Terrades et al. (2009) addressed the problem of combining classifier decisions using a non-Bayesian probabilistic framework.

Summarizing, the base classifiers in the ensemble construction methods are trained on different subsets of the training set. The approach toward compiling the training subsets differs among the ensemble generation algorithms. Fusion approaches aim to combine the decisions of the base classifiers. As the aim of ensemble construction methods is to identify difficult-to-classify training subsets, we are motivated to incorporate clustering in this regard as it partitions a classified data set into regions of highly correlated data.

3. Cluster-based ensemble classifier

The objective of the cluster-based ensemble classifier is to achieve diversity by explicit partitioning of the data set into multiple clusters and engaging a set of base classifiers to learn the decision boundaries within the patterns of the clusters. As the different base classifiers focus on different portion of the data set, the method achieves higher diversity. The theoretical framework underlying the ensemble classifier and the learning and prediction processes are detailed below in Sections 2.1 and 2.2, respectively.

3.1. Theoretical framework

We start this section with focus on clustering. Clustering refers to the possibility of existence of multiple clusters within a data set (Figure 1 (a)). While a classified data set is partitioned into multiple clusters, each cluster can be either atomic or non-atomic. An *atomic cluster* contains patterns from the single class (Figure 1 (b)) whereas a *non-atomic cluster* contains patterns from multiple classes (Figure 1 (b)). A pattern with unknown class that is close to an atomic cluster in Euclidian space takes the label of the patterns in that cluster. To classify a pattern that is close to non-atomic cluster, a decision boundary needs to be learned (Figure 1 (c)). As there are multiple non-atomic clusters within a data set, we bring in the ensemble classifier at this point. A set of base classifiers learn the decision boundaries among the patterns of different classes within each cluster. For classifying a pattern with unknown class, the cluster in which that pattern belongs is obtained first and the corresponding base classifier is used for predicting the class. As the base classifiers are focusing on different subsets of the training set, it achieves diversified learning of the base classifiers. The underlying theoretical framework for the overall process is detailed next.

Consider a set of training patterns $\Gamma = \{(\vec{x}_1, t_1), (\vec{x}_2, t_2), \ldots, (\vec{x}_{|\Gamma|}, t_{|\Gamma|})\}$ in a data set where each pattern is described by a vector of continuous valued features $\vec{x}_j = \langle x_{j1}, x_{j2}, \ldots, x_{jn}\rangle$ and a class label t_j with $t_j \in \{c_1, c_2, \ldots, c_{N_{class}}\}$. The representation of patterns can be considered as data points in a Euclidian space of dimension n. Clustering is the process of grouping data points that are geometrically close to each other. A distance function d between two patterns $(\vec{x}_i, t_i)$ and $(\vec{x}_j, t_j)$ in the training set is defined in terms of their Euclidian distance as

$$d(\vec{x}_i, \vec{x}_j) = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2} \tag{1}$$

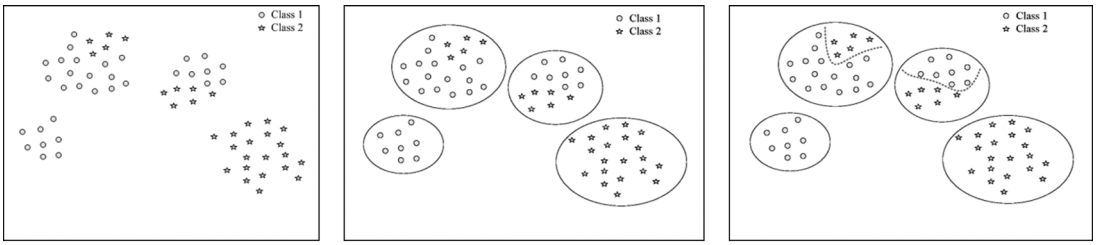
where $\vec{x}_i = \langle x_{i1}, x_{i2}, \ldots, x_{in}\rangle$ and $\vec{x}_j = \langle x_{j1}, x_{j2}, \ldots, x_{jn}\rangle$. Assuming a set of K clusters $\{\Omega_1, \Omega_2, \ldots, \Omega_K\}$ and the associated cluster centres $\{\vec{\kappa}_1, \vec{\kappa}_2, \ldots, \vec{\kappa}_K\}$ the clustering algorithm aims to minimize an objective function

$$J = \sum_{k=1}^{K} \sum_{\forall \vec{x}_i \in \Omega_k} d(\vec{x}_i, \vec{\kappa}_k) \tag{2}$$

for the data points in the training set Γ.

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Chua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) (b) (c)

Figure 1: *Impact of clustering on scattered data. (a) The data set with two classes. (b) Clustered data set with multiple atomic and non-atomic clusters. (c) Class boundaries learned on non-atomic clusters.*

After clustering a training example $(\vec{x}_i, t_i)$ can be augmented to $(\vec{x}_i, t_i, l_i)$ such that $\vec{x}_i \in \Omega_{l_i}$ where $1 \leq i \leq |\Gamma|$ and $1 \leq l_i \leq K$. For identification of atomic clusters, a class-cluster co-occurrence matrix φ can be computed as

$$\phi(q, r) = \sum_{\forall(\vec{x}_i, t_i, l_i)} \theta\left((t_i, q), (l_i, r)\right) \tag{3}$$

with

$$\theta\left((t, q), (l, r)\right) = \begin{cases} 1 & \textit{iff } q = t \text{ and } l = r \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $1 \leq q \leq N_{class}$ and $1 \leq r \leq K$. A cluster Ω_k can be identified as an atomic cluster iff

$$\frac{\sum_{\forall 1 \leq i \leq N_{class}} \phi(i, k)}{\max(\{\phi(i, k) \,|\, 1 \leq i \leq N_{class}\})} = 1 \tag{5}$$

Without loss of generality let us assume that $\{\Omega_1, \Omega_2, \ldots, \Omega_M\}$ are non-atomic clusters and $\{\Omega_{M+1}, \Omega_{M+2}, \ldots, \Omega_K\}$ are atomic clusters. A training subset Γ_j can be associated with a cluster Ω_j such that an example $(\vec{x}_i, t_i) \in \Gamma_j$ iff $j = l_i$ where $(\vec{x}_i, t_i, l_i)$ is the corresponding augmented example and $1 \leq i \leq |\Gamma|$ and $1 \leq j \leq K$. The class associated with an atomic cluster Ω_k can be computed as

$$\hat{t} = \arg\max_i \phi(i, k) \tag{6}$$

A set of base classifiers can be trained on the patterns of each of the non-atomic clusters. $\Gamma_1, \Gamma_2, \ldots, \Gamma_M$ represent the training subsets corresponding to non-atomic clusters $\Omega_1, \Omega_2, \ldots, \Omega_M$. Given a training subset Γ_i, the learning algorithm for a base classifier ψ_i finds out the decision boundaries among the patterns of the different classes represented by a set of classifier model parameters θ^{ψ_i} such that

$$\hat{\theta}^{\psi_i} = \arg\max_{\theta^{\psi_i}} \sum_{\forall(\vec{x}_j, t_j) \in \Gamma_i} \upsilon^{\psi_i}(\vec{x}_j, t_j) \tag{7}$$

where $\upsilon^{\psi_i}(\vec{x}_j, t_j)$ represents the separation capability function of the base classifier ψ_i over the examples $\{(\vec{x}_j, t_j)\}$ of the training set Γ_i such that

$$\upsilon^{\psi_i}(\vec{x}_j, t_j) = \begin{cases} +\alpha & \textit{if } \psi_i(\vec{x}_j | \hat{\theta}^{\psi_i}) = t_j \\ -\beta & \textit{otherwise} \end{cases} \tag{8}$$

where α, β are positive real numbers. A high the value of $\sum^{\upsilon^{\psi_i}}(\vec{x}_j, t_j)$ over the training examples represent better separability. The decision boundaries are later used for predicting the class of new examples.

Classification of a new pattern $\vec{x}$ starts by finding the appropriate cluster k as

$$k = \arg\min_k d(\vec{x}, \vec{\kappa}_k) \tag{9}$$

where d is the Euclidian distance defined in (1). If Ω_k is an atomic cluster, the class of the training patterns learned in (6) decides the fate of the new pattern. If Ω_k is a non-atomic cluster the corresponding base classifier ψ_k is used to predict the class of the new pattern in terms of the model parameters θ^{ψ_k} learned using (7).

3.2. Methodology

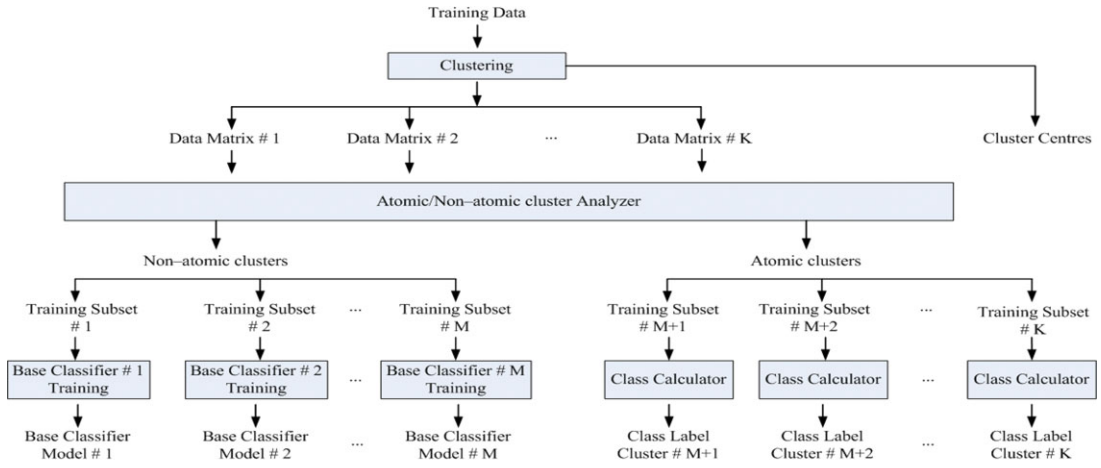
Based on the above framework, the learning and prediction processes for the cluster-based ensemble classifier are presented in Figure 2. During learning process the training data is partitioned into multiple clusters. The clusters are represented by their corresponding cluster centres. A cluster analyser then separates the computed clusters into atomic and non-atomic clusters. For atomic clusters, all the training patterns belong to the same class. A class label is computed for each of the atomic clusters. A non-atomic cluster contains patterns from multiple classes and a set of base classifiers are trained on each of the non-atomic clusters.

The class of a new pattern is predicted using the following steps. A test pattern is associated to the correct cluster by calculating the minimum of Euclidian distance between test pattern and the cluster centres. If the cluster is atomic in nature, the class label computed during training for the corresponding cluster is declared as the class of the test pattern. If the cluster is non-atomic in nature, the base classifier trained on the patterns of the corresponding non-atomic cluster is used to predict the class of the test pattern. The different components of the learning and prediction phases of the proposed ensemble classifier are presented next.

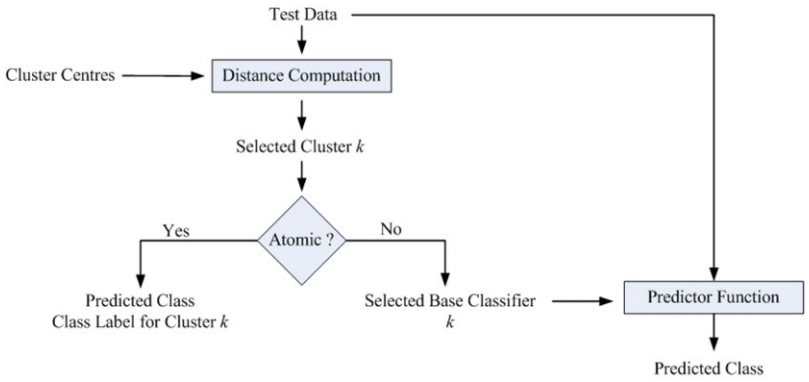
3.2.1. Clustering classified data The learning process starts by partitioning the training data into multiple clusters. Given the training data set $[x_{ij}] = [d_{ij}] \cdot [class_i]$ where $1 \leq i \leq N_{\text{examples}}$ and $1 \leq j \leq N_{\text{features}}$, the purpose of the clustering algorithm is to partition the training data into a number of N_{cluster} clusters. The output of the clustering algorithm is the modified data set $[y_{ij}] = [d_{ij}] \cdot [class_i] \cdot [cluster_i]$. Given the training data set, the clustering algorithm is presented in

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) Learning in Cluster based Ensemble Classifier



(b) Prediction in Cluster based Ensemble Classifier

Figure 2: *Cluster-based ensemble classifier. (a) Learning in cluster-based ensemble classifier. (b) Prediction in cluster-based ensemble classifier.*

Algorithm *Clustering_Classified_Data*

Input: Data Matrix $[d_{ij}]$, class vector $[class_i]$, # of clusters $N_{clusters}$.

Output: Modified data set $[y_{ij}] = [d_{ij}] \bullet [class_i] \bullet [cluster_i]$, cluster centres $\{\vec{\kappa}_1, \vec{\kappa}_2, \ldots, \vec{\kappa}_{N_{cluster}}\}$ where $[cluster_i]$ is the cluster vector.

1. A number of $N_{cluster}$ rows namely $\{\vec{\kappa}_1, \vec{\kappa}_2, \ldots, \vec{\kappa}_{N_{cluster}}\}$ are selected from $[d_{ij}]$ as initial cluster centres.
2. The distance between each row of $[d_{ij}]$ and the cluster centres $\{\vec{\kappa}_1, \vec{\kappa}_2, \ldots, \vec{\kappa}_{N_{cluster}}\}$ are computed.
3. Each row of $[d_{ij}]$ is assigned to a cluster whose centre is at minimum dissimilarity from that.
4. A new cluster centre is computed for each cluster from the data points belonging to that cluster.
5. Steps 2 and 3 are repeated until the cluster centres stop changing.

Figure 3: *Clustering algorithm for partitioning classified data into multiple clusters.*

Figure 3. At the completion of clustering each row of $[d_{ij}]$ is augmented with cluster id producing $[y_{ij}] = [d_{ij}] \cdot [cluster_i]$.

3.2.2. Atomic/non-atomic cluster analyser Given the modified training data $[y_{ij}] = [d_{ij}] \cdot [class_i] \cdot [cluster_i]$, the cluster analyser labels each cluster as either atomic or non-atomic. A cluster that consists of data belonging to only one class is labelled as atomic cluster. A cluster that does not satisfy this condition is labelled non-atomic cluster. The output of the cluster analyser is a column vector *cluster_status* where $cluster_status_j = 0$ for a non-atomic cluster and $cluster_status_j = 1$ for an atomic cluster. For the atomic cluster only the class label is remembered. The cluster analyser algorithm is presented in Figure 4.

3.2.3. Base classifier training A set of base classifiers are trained on the data sets of non-atomic clusters. The learning

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Algorithm *Cluster Analyser*

Input: Modified Training Data $[y_{ij}] = [d_{ij}] \bullet [class_i] \bullet [cluster_i]$, # of clusters $N_{clusters}$.

Output: Column vector *cluster_status* of size $N_{clusters}$.

```
1.  class_cluster_co_occur=zeros(N_classes,N_clusters);
2.  for i=1: |Γ|
3.          class_cluster_co_occur(class_i,cluster_i)+=1
4.  end for
5.  for i=1: N_clusters
6.          if max(class_cluster_co_occur(_,i))==sum(class_cluster_co_occur(_,i))
7.                  cluster_status(i)=1;
8.          else
9.                  cluster_status(i)=0;
10.         end if
11. end for
```

Figure 4: *Cluster analyser algorithm for finding atomic and non-atomic clusters.*

Algorithm *Base Classifier Training*

Input: Training Data $\Gamma_k = [d_{ij}] \bullet [class_i]$ corresponding to cluster k, the learning algorithm ϑ^k and learning goal $\mathcal{G}$.

Output: Base classifier model $\hat{\theta}^k$.

1. Initialize the model parameters $\hat{\theta}^k$ for the base classifier k.
2. Compute class labels of the training examples using $\hat{\theta}^k$.
3. Compute $\sum v^k(\vec{x}_j, t_j)$ as in (7) over the computed class labels produced from the training examples.
4. Adjust the model parameters $\hat{\theta}^k$ as guided by the learning algorithm ϑ^k.
5. Repeat steps 2 – 4 until the learning goal $\mathcal{G}$ is achieved.

Figure 5: *Base classifier training algorithm.*

algorithm for a generic base classifier is presented in Figure 5. The model parameters $\hat{\theta}^{\vec{k}}$ of the base classifier corresponding to cluster k is initialized to some random values. The learning algorithm ϑ^k then tunes the model parameters to maximize the objective function in equation (7) until some learning goal is achieved.

3.2.4. Prediction As the test example $e = \langle e_1, \ldots, e_{N_{\text{features}}} \rangle$ is presented to the prediction algorithm it finds the appropriate cluster from the distances with the pre computed cluster centres. If the associated cluster is atomic, the unique class of data points within the cluster is returned. For a non-atomic class, the corresponding base classifier predicts the class. The algorithm for prediction is presented in Figure 6.

4. Results and discussion

4.1. Experimental platform

We have conducted a number of experiments on benchmark data sets. We have compiled the data sets as used in contemporary research works (Rodriguez & Maudes, 2008; Munoz *et al.*, 2009; Pedrajas, 2009) from UCI Machine Learning Repository (UCI Machine Learning Database, 2012). A summary of the data sets is presented in Table 1. We used 10-fold cross-validation approach for reporting the results. Clustering of the data sets were done using the *k*-means clustering algorithm. We have used *k*-NN, NN, DT and SVM (Kim *et al.*, 2002) as the base classifiers in the different experiments. We used libsvm library (LIBSVM, 2012) and radial basis kernel for SVM while *tansigmoid* transfer function

Algorithm *Prediction*

Input: Test Example $e =< e_1, \ldots, e_{N_{features}} >$.

Output: Predicted Class *class_label*.

1. The distance between e and all the cluster centres $\{\vec{\kappa}_1, \vec{\kappa}_2, \ldots, \vec{\kappa}_{N_{cluster}}\}$ and select the k–th cluster for which the distance is minimum.

```
2.  if cluster_status(k)==1
3.          class_label=The class of the training examples of atomic cluster k.
4.  else
5.          class_label=class predicted by the base classifier of non–atomic cluster k.
6.  end if
```

Figure 6: *Ensemble classifier prediction algorithm.*

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 1: *Data sets used in the experiments*

Data set	*No. of instances*	*No. of attributes*	*No. of classes*	*Test process*
Breast cancer	699	10	2	10-fold cross-validation
Diabetes	768	8	2	10-fold cross-validation
E. coli	336	7	8	10-fold cross-validation
German	1000	20	2	10-fold cross-validation
Glass	214	10	7	10-fold cross-validation
Ionosphere	351	34	2	10-fold cross-validation
Iris	150	4	3	10-fold cross-validation
Liver	345	6	2	10-fold cross-validation
Parkinsons	197	23	2	10-fold cross-validation
Satellite	6435	36	6	10-fold cross-validation
Segment	2310	19	7	10-fold cross-validation
Sonar	208	60	2	10-fold cross-validation
Spect	267	23	2	10-fold cross-validation
Thyroid	215	5	3	10-fold cross-validation
Transfusion	748	5	2	10-fold cross-validation
Vehicle	946	18	4	10-fold cross-validation
Vowel	528	10	11	10-fold cross-validation
Waveform	5000	21	3	10-fold cross-validation
Wine	178	13	3	10-fold cross-validation
Yeast	1484	8	10	10-fold cross-validation

was used with NN in the experiments. A part of the training data was used as validation set for optimization of the classifier parameters namely k in k-NN classifier, "Hidden Units" in NN and *sigma* of radial basis function (RBF) kernel in SVM. The classification results of bagging and boosting are compared with the cluster-based ensemble classifier and the results are obtained using WEKA (Hall *et al*., 2009). All the experiments were conducted on MATLAB 7.5.0.

4.2. Experimental results

The discussions with graphical representations in first few sections are confined to a subset of data sets in Table 1. A discussion on accuracies and comparative analysis on all the data sets is presented at the end of this section.

4.2.1. Atomic/non-atomic clusters Partitioning a data set creates multiple atomic and non-atomic clusters as evidenced from the data sets. Figure 7 represents the class-cluster co-occurrence matrices when the data sets are partitioned into three clusters. Both atomic and non-atomic clusters are created for some data sets. As evidenced from Figure 7, Cluster 3 in Breast cancer (Class 2), Cluster 1 in Iris (Class 1), Cluster 3 in Sonar (Class 2), Cluster 2 in Segment (Class 2), Cluster 1 and Cluster 3 in Thyroid (Class 2 and Class 3, respectively) are atomic whereas the remaining clusters for these data sets are non-atomic as they contain instances from multiple classes. Only the class label is remembered for an atomic cluster whereas base classifiers are trained on non-atomic clusters. Note that the number of base classifiers depends on number of non-atomic clusters and it varies from one data set to another. Partitioning some data sets such as Ionosphere created no atomic clusters.

Figure 8 represents the number of atomic and non-atomic clusters when different data sets are partitioned into multiple clusters. Note that the proportion of atomic and non-atomic clusters is different for different data sets. Higher numbers of atomic clusters are created compared to non-atomic clusters for Iris and Thyroid data sets. The scenario is opposite for data sets such as Sonar, that is, the proportion of non-atomic

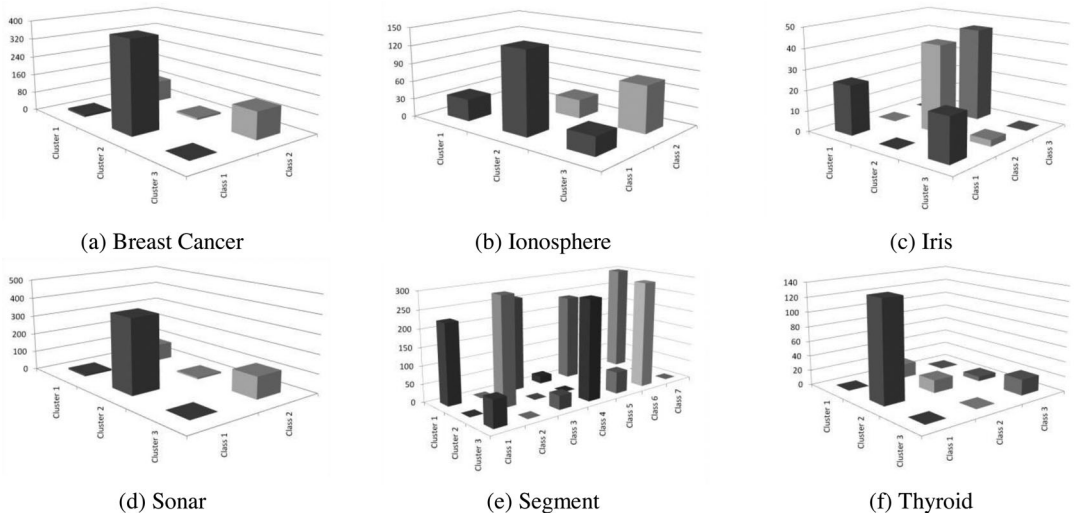
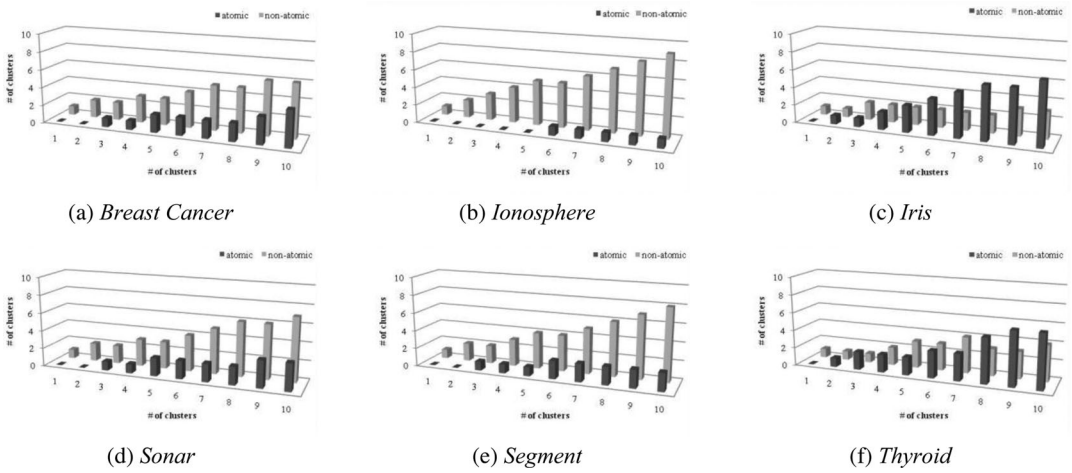


Figure 7: *Atomic and non-atomic clusters while the training data corresponding to the first fold of the data sets in Table 1 are partitioned into three clusters.*

© 2012 Wiley Publishing Ltd

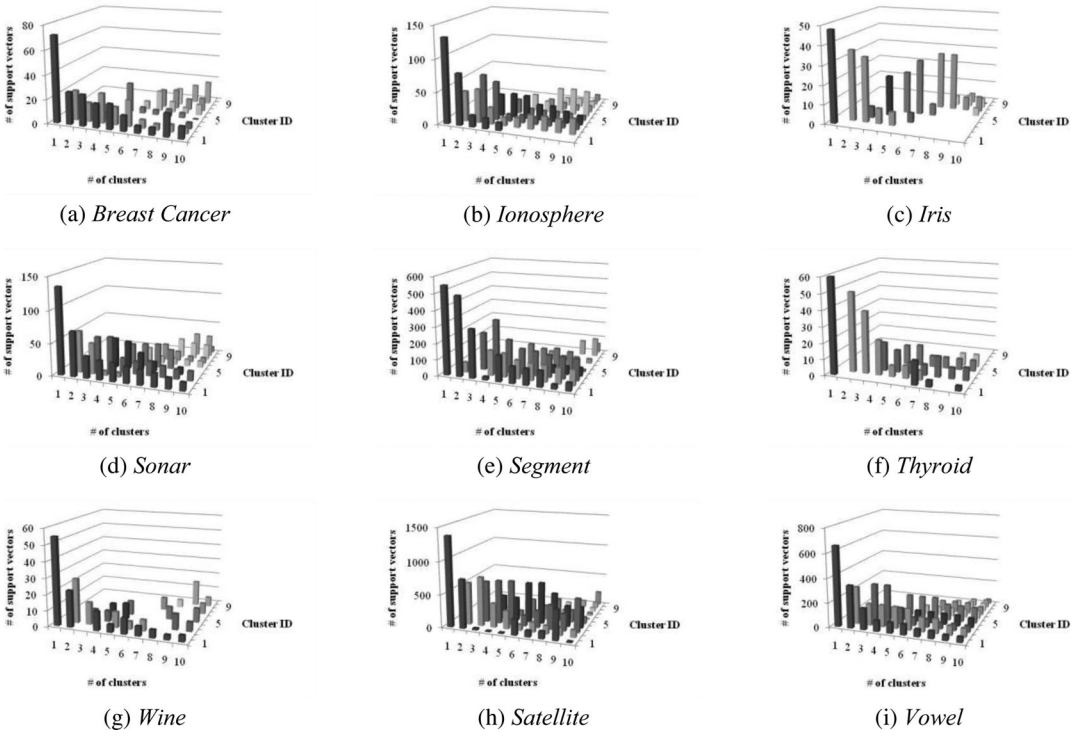
14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

Figure 8: *Proportion of atomic vs non-atomic clusters when the training data corresponding to the first fold of the data sets in Table 1 are partitioned into multiple clusters.*



(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

(g) *Wine* (h) *Satellite* (i) *Vowel*

Figure 9: *Number of support vectors at different number of clusters for the data sets in Table 1 at which the maximum classification accuracy is achieved corresponding to fold one using support vector machine (SVM) classifier.*

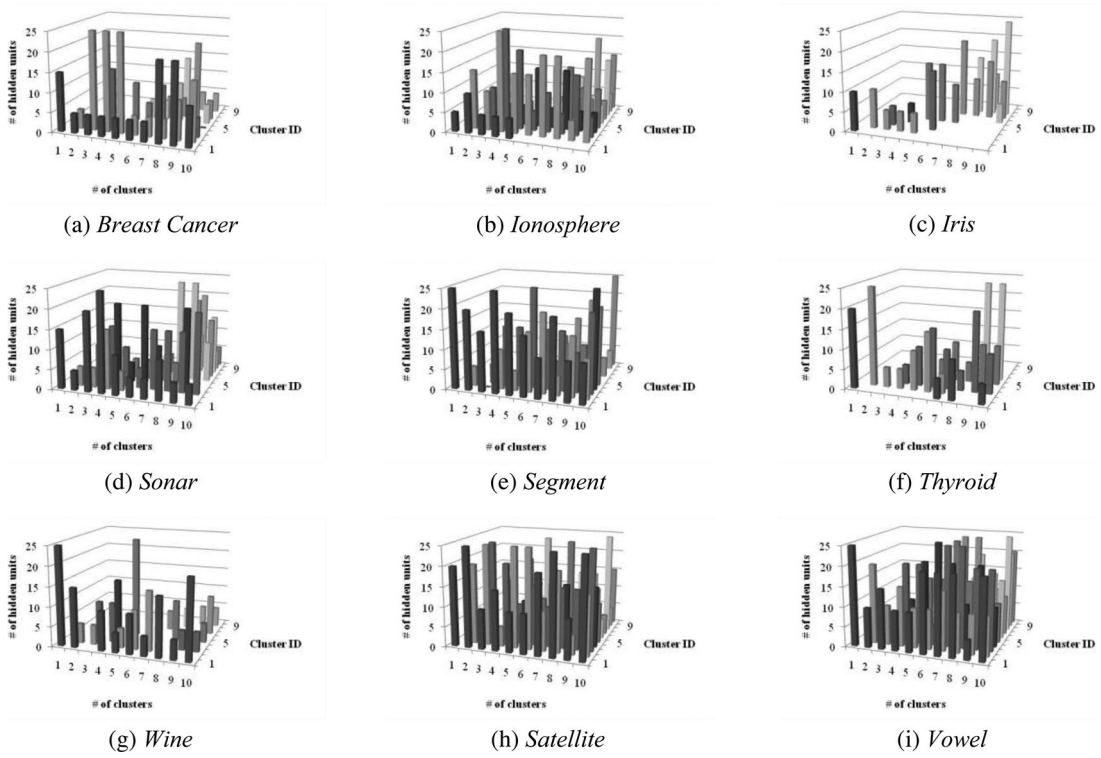
clusters is higher. There are a number of implications of these proportions:

(1) As only the class label needs to be remembered for the atomic clusters, a higher proportion of atomic clusters indicate lower memory/storage requirement. On the other hand, a higher number of non-atomic clusters indicate the necessity of using more base classifiers for learning and thus require higher storage/memory as the corresponding parameters are required to be remembered for later prediction.
(2) Higher number of non-atomic clusters implies higher training time as more base classifiers are needed to be trained.
(3) A higher proportion of atomic cluster means that highly correlated data points belong to the same class. These types of data sets are in general well separable and thus higher classification accuracy is expected for these data sets compared to others. On the other hand, for the data sets with lower proportion of atomic clusters, the classification accuracy depends on how well the base classifiers learn the decision boundaries.

4.2.2. Impact of clustering on classifier learning This section presents some analysis on the influence of clustering on classifier learning parameters. The parameters under consideration are number of hidden units in NN, number of support vectors in SVM and the value of k in k-NN classifier. Figure 9 represents the number of support vectors required to learn the decision boundary for the different clusters obtained by partitioning the data sets. In general, the number of support vectors is reduced as the data are partitioned

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

(g) *Wine* (h) *Satellite* (i) *Vowel*

Figure 10: *Number of hidden units at different number of clusters for the data sets in Table 1 at which the maximum classification accuracy is achieved corresponding to fold one using neural network (NN) classifier.*

in higher number of clusters. The number of training examples is reduced in each cluster with a higher number of partitioning and the decision boundaries become easy to learn and thus reduced numbers of support vectors are required.

The number of hidden units for NN training as required at the different number of clusters is presented in Figure 10. Data sets such as Wine, Iris and Thyroid have relatively well-defined decision boundaries among the classes. Relatively smaller numbers of hidden units are required for most of their clusters. Data sets such as Satellite require higher number of hidden units for learning the clusters. As indicated in Table 1 Satellite data set has higher number of instances.

The k values at which the k-NN classifier performs best on the different clusters of the validation set is presented in Figure 11. In general, the value of k is mostly limited to one for small as well as large data sets. Sometimes higher values of k (e.g. three, five etc.) are found to perform better in some clusters for data sets with large number of instances.

4.2.3. Impact of clustering on classification accuracy The experimental results that we have presented here aim to find the number of clusters at which the base classifiers achieve the highest classification accuracy. Figures 12–14 summarize the classification results achieved by the ensemble classifier at different number of clusters for the data sets in Table 1 using SVM, NN and k-NN as the base classifiers, respectively. In all cases, the classification accuracy for the original data set is represented by only one cluster for the entire data set. Note that for most of the data sets the highest classification accuracy is achieved at more than one cluster. This indicates the fact that classification accuracy improves by learning decision boundaries through clusters. The cluster wise learning of the base classifiers improves the learning domain resulting in better performance of the ensemble classifier.

The learning process of different classifiers is dissimilar. Thus, the accuracy achieved at different number of clusters is also different as noted from the accuracy curves in Figures 12–14 for different classifiers. Figure 12 represents the classification accuracies using SVM classifier. The best classification accuracy for Sonar data set is obtained at one cluster whereas that for Iris, Vowel, and Wine data set is achieved at two clusters. Thyroid data set achieves the best accuracy at three clusters. The classification accuracy peaks at six clusters for Ionosphere and Breast cancer data sets. Best classification accuracy for Satellite and Segment data sets is obtained at seven and nine clusters, respectively.

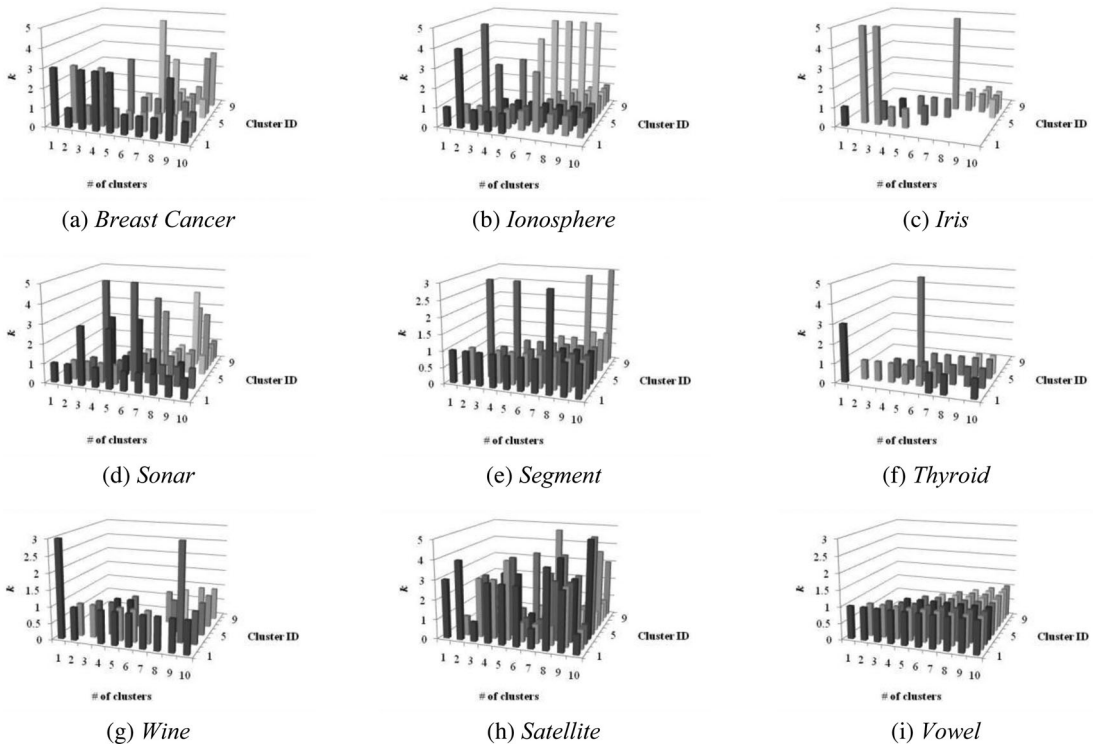
Classification accuracies obtained using NN classifier is presented in Figure 13. The best accuracy for Satellite data set is obtained using at one cluster. The Segment data set peaks at two clusters. Wine data set attains the maximum classification accuracy at three clusters. Best classification accuracy for Ionosphere and Vowel data sets is obtained at four clusters. The maximum accuracy for Breast cancer and Sonar data sets are achieved at five clusters. Thyroid and Iris data sets reach the best accuracies at six and nine clusters, respectively.

With k-NN classifier (Figure 14), the best accuracy for Segment, Satellite and Wine data sets are obtained at one cluster. The accuracy attains the maximum at three clusters for Iris data set. The Breast cancer and Ionosphere data sets get the maximum at four clusters whereas for Vowel data set the classification accuracy peaks at seven clusters. The best accuracies for Sonar and Thyroid data sets are obtained at eight and 10 clusters, respectively.

It is clear from the above analysis that there is no straightforward correlation between the number of clusters and the

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

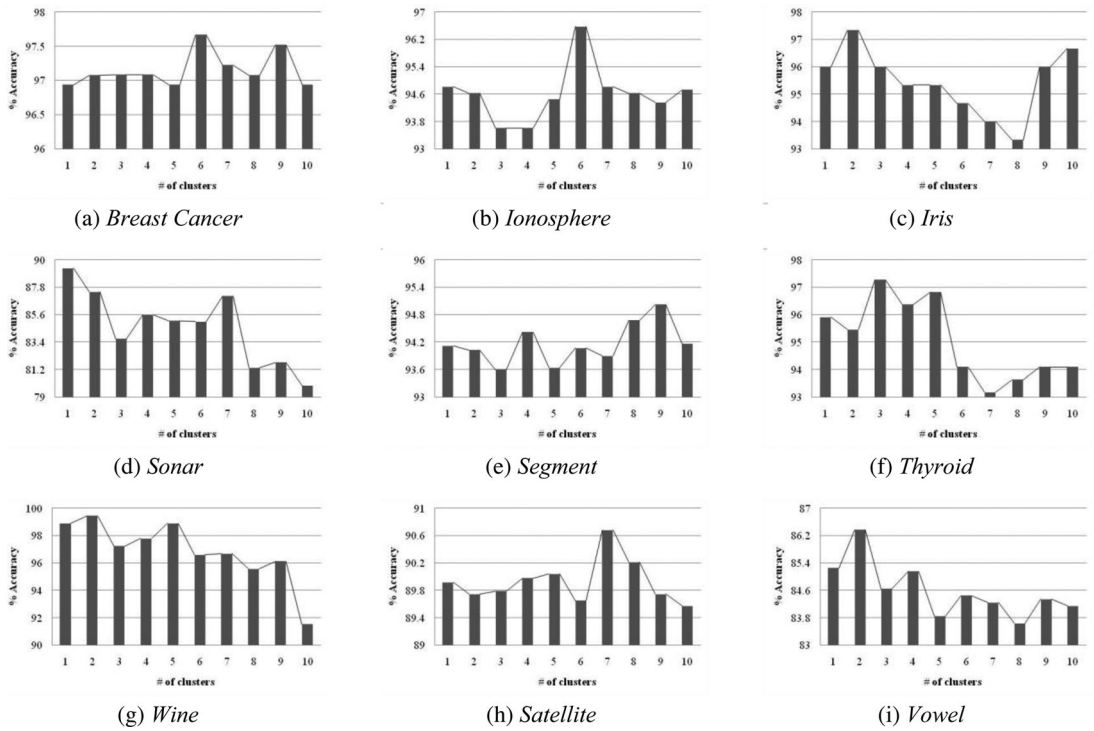


(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

(g) *Wine* (h) *Satellite* (i) *Vowel*

Figure 11: *k values at different number of clusters for the data sets in Table 1 at which the maximum classification accuracy is achieved corresponding to fold one using k-NN classifier.*



(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

(g) *Wine* (h) *Satellite* (i) *Vowel*

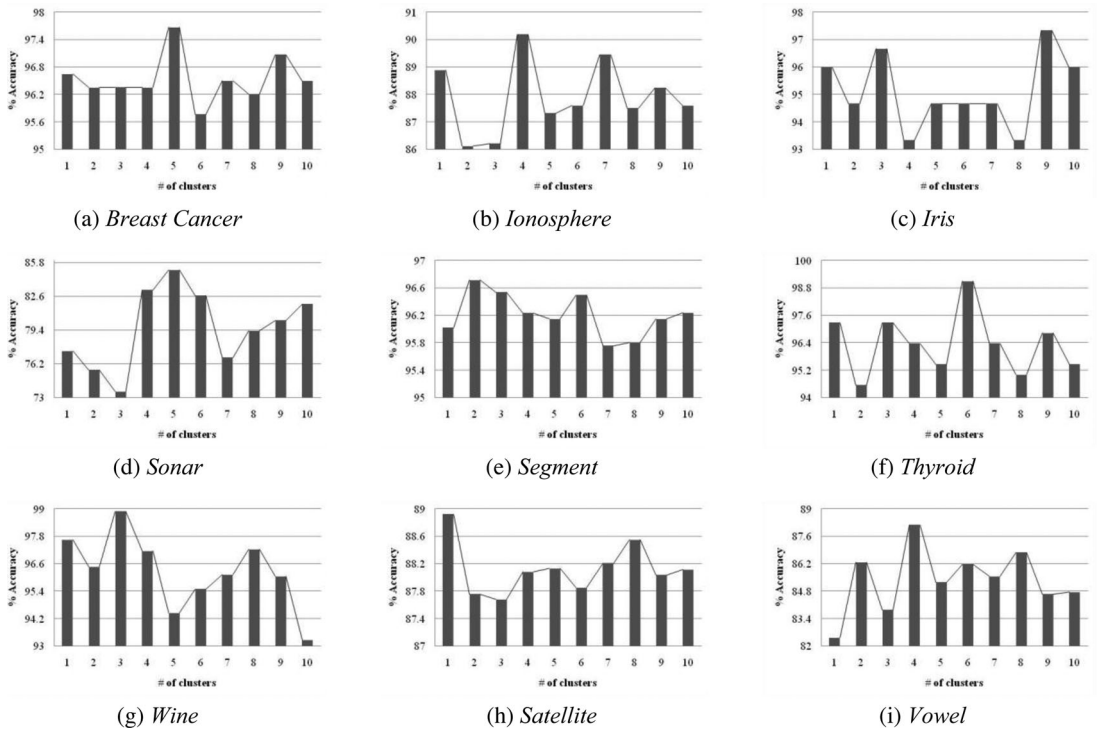
Figure 12: *Classification accuracies at different number of clusters for the data sets in Table 1 using support vector machine (SVM) classifier.*

classification accuracy. The number of clusters at which the maximum accuracy is achieved for all the data sets using *k*-NN, NN, SVM and DT classifiers is reported in Table 2. It is interesting to observe from Table 2 that different base classifiers achieve maximum accuracy at different number of clusters.

4.2.4. Comparative analysis A comparative picture of maximum accuracies achieved using the four base classifiers is presented in Table 3. It is interesting to observe that the highest accuracy varies from classifier to classifier and no single classifier achieves the maximum for all the data sets in Table 1. SVM achieves maximum accuracy for 10 data sets

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

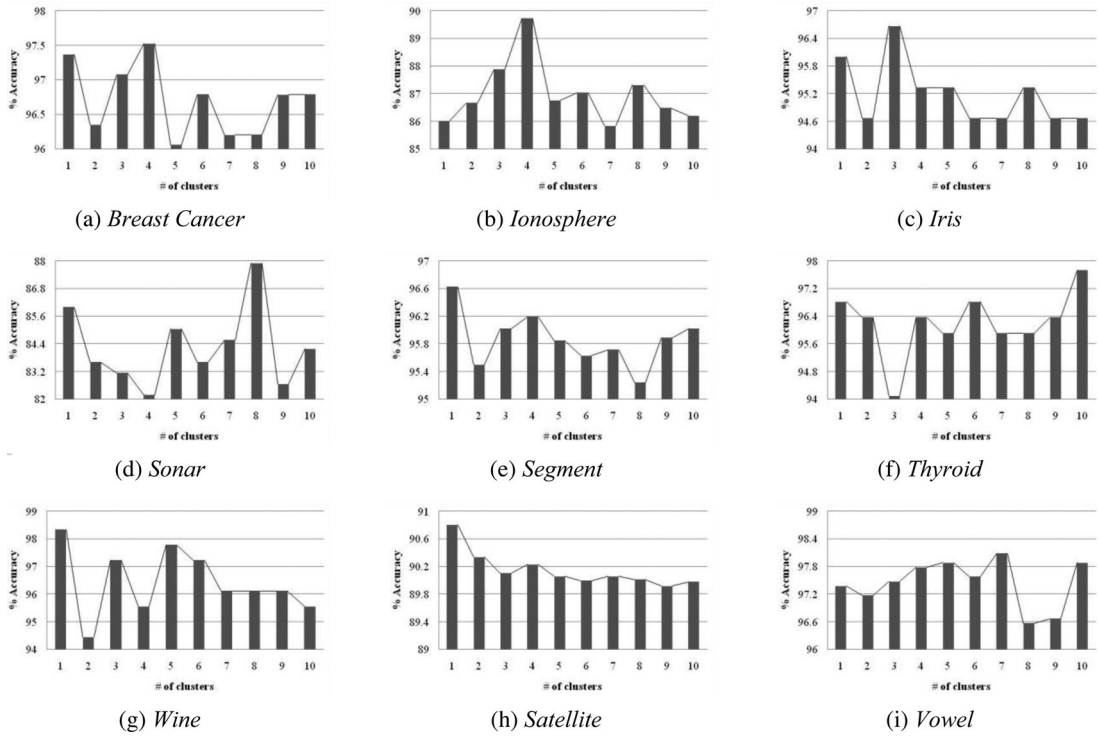


(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

(g) *Wine* (h) *Satellite* (i) *Vowel*

Figure 13: *Classification accuracies at different number of clusters for the data sets in Table 1 using neural network (NN) classifier.*



(a) *Breast Cancer* (b) *Ionosphere* (c) *Iris*

(d) *Sonar* (e) *Segment* (f) *Thyroid*

(g) *Wine* (h) *Satellite* (i) *Vowel*

Figure 14: *Classification accuracies at different number of clusters for the data sets in Table 1 using k-NN classifier.*

including Breast cancer, Diabetes, German, Ionosphere, Iris, Sonar, Spect, Waveform, Wine and Yeast data sets. NN obtains the maximum accuracy for seven data sets including Breast cancer, Iris, Liver, Segment, Thyroid, Transfusion, and Vehicle data sets. Maximum accuracy using DT was obtained on three data sets including E. coli, Glass, and Segment data sets. Using *k*-NN classifier, the highest accuracy is attained for three data sets namely Parkinsons, Satellite and Vowel. Overall cluster-based ensemble with SVM base classifier performs 1.88%, 0.48%, and 4.01% better than that with *k*-NN, NN, and DT base classifiers, respectively.

We have provided a comparison between the best performing cluster ensemble (i.e. with SVM base classifier) with some of the contemporary ensemble classifiers namely Bagging and Boosting. The classification accuracies obtained on the 20 data sets are reported in Table 4. Overall clustered ensemble with SVM base classifier performs 2.79% and 3.73% better than bagging and boosting in terms of average

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 2: *Number of clusters at which the highest classification accuracy was achieved for data sets in Table 1 using different classifiers*

Data set	*k-NN*	*NN*	*SVM*	*DT*	*Data set*	*k-NN*	*NN*	*SVM*	*DT*
Breast cancer	4	5	6	8	Segment	1	2	9	8
Diabetes	1	2	3	2	Sonar	8	5	1	4
E. coli	1	8	7	10	Spect	1	1	1	1
German	6	1	3	1	Thyroid	10	6	3	6
Glass	1	1	1	1	Transfusion	3	1	6	6
Ionosphere	4	4	6	3	Vehicle	2	1	2	2
Iris	3	9	10	9	Vowel	7	4	2	10
Liver	8	2	1	6	Waveform	5	9	1	2
Parkinsons	8	4	5	5	Wine	1	3	2	2
Satellite	1	1	7	9	Yeast	1	2	1	8

Table 3: *Average classification accuracy on the data sets in Table 1 using k-NN, neural network (NN) and support vector machine (SVM) as the classifier. Maximum accuracies marked in bold*

Data set	*Ensemble k-NN*	*Ensemble NN*	*Ensemble SVM*	*Ensemble DT*
Breast cancer	97.52 ± 2.16	**97.67 ± 1.39**	**97.67 ± 1.39**	96.06 ± 2.03
Diabetes	74.23 ± 3.67	77.07 ± 4.43	**77.73 ± 2.80**	72.13 ± 3.16
E. coli	90.07 ± 4.85	87.50 ± 4.26	88.57 ± 5.69	**90.52 ± 3.05**
German	71.20 ± 3.85	73.50 ± 2.92	**74.60 ± 4.20**	71.80 ± 3.97
Glass	93.18 ± 5.77	92.73 ± 6.50	91.73 ± 4.21	**98.18 ± 3.18**
Ionosphere	89.72 ± 4.15	90.19 ± 6.90	**96.57 ± 4.10**	88.89 ± 5.71
Iris	96.67 ± 3.51	**97.33 ± 3.44**	**97.33 ± 3.44**	94.67 ± 4.22
Liver	67.00 ± 6.26	**72.19 ± 7.48**	70.00 ± 7.41	65.62 ± 8.42
Parkinsons	**95.81 ± 5.11**	92.93 ± 5.63	90.71 ± 5.06	89.14 ± 6.70
Satellite	**90.80 ± 0.97**	88.92 ± 0.86	90.68 ± 1.30	86.45 ± 1.06
Segment	96.62 ± 1.04	**96.71 ± 1.28**	95.02 ± 1.57	**96.71 ± 1.31**
Sonar	87.89 ± 6.33	85.09 ± 6.13	**89.80 ± 7.88**	80.80 ± 12.4
Spect	78.98 ± 6.47	82.18 ± 5.61	**84.72 ± 4.26**	83.61 ± 5.13
Thyroid	97.73 ± 3.21	**99.09 ± 1.92**	97.27 ± 2.35	95.46 ± 3.71
Transfusion	72.18 ± 5.56	**79.14 ± 3.19**	78.37 ± 2.02	75.70 ± 2.18
Vehicle	71.39 ± 2.55	**81.31 ± 5.15**	75.17 ± 2.18	70.66 ± 4.15
Vowel	**98.08 ± 1.30**	88.18 ± 4.97	86.36 ± 3.73	82.83 ± 5.30
Waveform	81.33 ± 3.22	82.67 ± 4.66	**87.00 ± 5.97**	78.00 ± 9.32
Wine	98.33 ± 3.75	98.89 ± 3.51	**99.44 ± 1.76**	96.11 ± 5.27
Yeast	58.36 ± 3.51	60.30 ± 4.33	**62.19 ± 3.94**	57.00 ± 4.48

classification accuracy. Clustered ensemble outperforms bagging and boosting on 16 and 18 data sets, respectively. This implies that clustered ensemble with SVM base classifier performs significantly better than bagging and boosting. In order to validate the claims, we define the null and alternative hypothesis for each classifier pair in Table 5. Note that the null hypothesis is rejected at 0.05 significance level by two-tailed sign test for each classifier pair in Table 5 implying that

Table 4: *Comparative classification accuracy (%) between clustered ensemble and contemporary ensemble classifiers*

Data set	*Clustered ensemble SVM*	*Bagging*	*Boosting*
Breast cancer	97.67 ± 1.39	96.64 ± 2.08	94.00 ± 1.65
Diabetes	77.73 ± 2.80	76.56 ± 2.64	74.48 ± 4.07
E. coli	88.57 ± 5.69	87.40 ± 4.62	85.82 ± 3.78
German	74.60 ± 4.20	75.60 ± 3.60	73.20 ± 4.69
Glass	91.73 ± 4.21	97.27 ± 3.83	97.73 ± 4.66
Ionosphere	96.57 ± 4.10	90.46 ± 5.47	91.76 ± 5.83
Iris	97.33 ± 3.44	94.67 ± 4.22	94.66 ± 2.67
Liver	70.00 ± 7.41	66.29 ± 5.18	64.81 ± 4.14
Parkinsons	90.71 ± 5.06	89.85 ± 7.01	83.74 ± 7.25
Satellite	90.68 ± 1.30	89.22 ± 1.08	88.66 ± 1.08
Segment	95.02 ± 1.57	96.54 ± 1.24	96.84 ± 0.99
Sonar	89.80 ± 7.88	80.43 ± 9.29	79.27 ± 6.92
Spect	84.72 ± 4.26	82.92 ± 6.71	82.13 ± 6.65
Thyroid	97.27 ± 2.35	94.55 ± 5.16	95.45 ± 2.87
Transfusion	78.37 ± 2.02	78.26 ± 4.72	78.08 ± 2.40
Vehicle	75.17 ± 2.18	71.72 ± 5.14	73.63 ± 4.22
Vowel	86.36 ± 3.73	85.15 ± 4.95	83.64 ± 3.16
Waveform	87.00 ± 5.97	74.67 ± 5.02	77.00 ± 8.49
Wine	99.44 ± 1.76	95.49 ± 4.41	95.00 ± 3.89
Yeast	62.19 ± 3.94	62.88 ± 4.18	61.66 ± 3.11

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Chua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 5: *Pairwise classification performance comparison between the cluster ensemble classifier (support vector machine, SVM) and the contemporary ensemble classifiers using two-tailed Sign test (Demsar, 2006; Sheskin, 2011)*

Classifier pair	*Hypothesis test*
Cluster ensemble (SVM) vs bagging	Null hypothesis H_0: The cluster ensemble method (SVM) is equivalent to bagging. Alternative hypothesis H_1: The cluster ensemble method (SVM) is significantly better than clustered ensemble. Sign hest: The cluster ensemble method (SVM) outperforms bagging on 16 data sets (Table 4). The critical value at $\alpha = 0.05$ is 15. Thus, the Null hypothesis H_0 is rejected at significance level $\alpha = 0.05$.
Cluster ensemble (SVM) vs boosting	Null hypothesis H_0: The cluster ensemble method (SVM) is equivalent to boosting. Alternative hypothesis H_1: The cluster ensemble method (SVM) is significantly better than boosting. Sign test: The cluster ensemble method (SVM) outperforms boosting on 18 data sets (Table 4). The critical value at $\alpha = 0.05$ is 15. Thus, the Null hypothesis $H_0$is rejected at significance level $\alpha = 0.05$.

cluster ensemble (with SVM base classifier) performs significantly better than bagging and boosting. The better performance can be attributed to the explicit finding of difficult-to-classify patterns by the cluster-based ensemble classifier.

5. Conclusion

In this paper, we have presented cluster-based ensemble classifier that partitions the data set into multiple atomic and non-atomic clusters and engages a set of base classifiers to learn the decision boundaries within them thus achieving diversity. Based on the experimental results, it can be concluded that the learning of class boundaries by the base classifiers through clusters achieves diversified learning leading to better classification accuracy compared to learning from the original data set with no clustering. There is no straightforward correlation between the number of clusters and the classification accuracy. The highest classification accuracy and the number of cluster at which it is achieved depend on the use of SVM, NN, DT and *k*-NN classifiers for a data set. The comparative results show that the performance of the cluster-based ensemble classifier (SVM) on a number of benchmark data sets is on an average 2.79% better accuracy than bagging and 3.73% better than boosting. A two-tailed sign test justifies that the improvement over bagging and boosting is significant. In future, we would also like to investigate the impact of incorporating overlapping clusters and using multiple base classifiers to learn and classify overlapping patterns.

References

Breiman, L. (1996) Bagging predictors, *Machine Learning*, **24**(2), 123–140.

Cevikalp, H. and R. Polikar (2008) Local classifier weighting by quadratic programming, *IEEE Transaction on Neural Networks*, **19**(10), 1832–1838.

Chen, L. and M.S. Kamel (2009) A generalized adaptive ensemble generation and aggregation approach for multiple classifiers systems, *Pattern Recognition*, **42**, 629–644.

Demsar, J. (2006) Statistical comparisons of classifiers over multiple data sets, *Journal of Machine Learning Research*, **7**, 1–30.

Freund, Y. and R.E. Schapire (1997) Decision-theoretic generalization of on-line learning and an application to boosting, *Journal of Computer and System Sciences*, **55**(1), 119–139.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I.H. Witten (2009) The WEKA data mining software: an update, *SIGKDD Explorations*, **11**(1), 10–18.

Kim, H.-C., S. Pang, H.-M. Je, D. Kim and S.Y. Bang (2002) Pattern classification using support vector machine ensemble, *International Conference on Pattern Recognition*, **2**, 160–163.

Kittler, J., M. Hatef, R.P.W. Duin and J. Matas (1998) On combining classifiers, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **20**(3), 226–239.

Ko, A.H.R., R. Sabourin, A. de S. Britto and L. Oliveira (2007) Pairwise fusion matrix for combining classifiers, *Pattern Recognition*, **40**, 2198–2210.

Kuncheva, L.I. (2000) Cluster-and-selection method for classifier combination, in Proceedings of the International Conference on Knowledge-Based Intelligent Engineering Systems & Allied Technologies (KES), 185–188.

Kuncheva, L.I., J.C. Bezdek and R. Duin (2001) Decision templates for multiple classifier fusion: an experimental comparison, *Pattern Recognition*, **34**(2), 299–314.

Le, C.A., V.-N. Huynh, A. Shimazua and Y. Nakamoria (2007) Combining classifiers for word sense disambiguation based on Dempster–Shafer theory and OWA operators, *Data & Knowledge Engineering*, **63**(2), 381–396.

LIBSVM (2012) *A Library for Support Vector Machines*, http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

Muhlbaier, M.D., A. Topalis and R. Polikar (2009) Learn++.NC: combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes, *IEEE Transaction on Neural Networks*, **20**(1), 152–168.

Munoz, G.M., D.H. Lobato and A. Suarez (2009) An analysis of ensemble pruning techniques based on ordered aggregation, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **31**(2), 245–259.

Nanni, L. and A. Lumini (2006) Fuzzy bagging: a novel ensemble of classifiers, *Pattern Recognition*, **39**, 488–490.

Nasierding, G., G. Tsoumakas and A.Z. Kouzani (2009) Clustering based multi-label classification for image annotation and retrieval, in Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 4514–4519.

Parikh, D. and R. Polikar (2007) Ensemble based incrimental learning approach to data fusion, *IEEE Transaction on Systeams, Man, and Cybernetics*, **37**(2), 437–450.

Pedrajas, N.G. (2009) Constructing ensembles of classifiers by means of weighted instance selection, *IEEE Transaction on Neural Networks*, **20**(2), 258–277.

Polikar, R. (2006) Ensemble based systems in decision making, *IEEE Circuits and Systems Magazine*, **6**(3), 21–45.

Rodriguez, J.J. and J. Maudes (2008) Boosting recombined weak classifiers, *Pattern Recognition Letters*, **29**, 1049–1059.

Rokach, L., O. Maimon and I. Lavi (2003) Space decomposition in data mining: a clustering approach, in *Proceedings of the International Symposium On Methodologies For Intelligent Systems*, 24–31.

Saffari, A., C. Leistner and H. Bischof (2009) Regularized multi-class semi-supervised boosting, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 967–974.

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

SCHAPIRE, R.E. (1990) The strength of weak learnability, *Machine Learning*, **5**(2), 197–227.

SHESKIN, D.J. (2011) *Handbook of Parametric and Nonparametric Statistical Procedures*, 5th edn, Boca Raton.

TANG, B., M.I. HEYWOOD and M. SHEPHERD (2002) Input partitioning to mixture of experts, in Proceedings of the International Joint Conference on Neural Networks, 227–232.

TERRADES, O.R., E. VALVENY and S. TABBONE (2009) Optimal classifier fusion in a non-Bayesian probabilistic framework, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **31**(9), 1630–1644.

TODOROVSKI, L. and S. DZEROSKI (2003) Combining classifiers with meta decision trees, *Machine Learning*, **50**, 223–249.

TSOUMAKAS, G., L. ANGELIS and I. VLAHAVAS (2004) Clustering classifiers for knowledge discovery from physically distributed databases, *Data & Knowledge Engineering*, **49**(3), 223–242.

UCI Machine Learning Database (2012) http://archive.ics.uci.edu/ml/.

WANAS, N.M., R.A. DARA and M.S. KAMEL (2006) Adaptive fusion and co-operative training for classifier ensembles, *Pattern Recognition*, **39**, 1781–1794.

XIUPING, J. and J.A. RICHARDS (2003) Cluster-space classification: a fast k-nearest neighbour classification for remote sensing hyperspectral data, in *Proceedings of the IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, 407–410.

© 2012 Wiley Publishing Ltd

14680394, 2013, 3, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2012.00637.x by Cbua-Consorcio De Bibliotecas, Wiley Online Library on [30/05/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License