Cluster-Based Cumulative Ensembles

Hanan G. Ayad and Mohamed S. Kamel

Pattern Analysis and Machine Intelligence Lab,
Electrical and Computer Engineering, University of Waterloo,
Waterloo, Ontario N2L 3G1, Canada
{hanan, mkamel}@pami.uwaterloo.ca
http://pami.uwaterloo.ca/

Abstract. In this paper, we propose a cluster-based cumulative representation for cluster ensembles. Cluster labels are mapped to incrementally accumulated clusters, and a matching criterion based on maximum similarity is used. The ensemble method is investigated with bootstrap re-sampling, where the k-means algorithm is used to generate high granularity clusterings. For combining, group average hierarchical *meta-clustering* is applied and the Jaccard measure is used for cluster similarity computation. Patterns are assigned to combined meta-clusters based on estimated cluster assignment probabilities. The cluster-based cumulative ensembles are more compact than co-association-based ensembles. Experimental results on artificial and real data show reduction of the error rate across varying ensemble parameters and cluster structures.

1 Introduction

Motivated by the advances in classifier ensembles, which combine the predictions of multiple classifiers; cluster ensembles that combine multiple data partitionings have started to gain an increasing interest [1–8].

Cluster ensembles can be illustrated by the schematic model in Figure 1. The model includes two main elements, the ensemble generation and the combination scheme. The ensemble generation takes as input a dataset of d-dimensional pattern vectors represented by an $N \times d$ matrix $\mathbf{X} = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$, where N is the number of patterns and the row vector $\mathbf{x}^{(i)}$ represents the ith pattern. The ensemble generation generates multiple clusterings, represented here by cluster label vectors $\{\mathbf{y}^{(b)}\}_{b=1}^{B}$. The combining scheme (or the consensus function [1]), can be thought of as comprising two sub-elements. The first is the ensemble mapping which defines a representation $\mathbf{Z}$ of the ensemble outputs and an associated mapping method. The lack of direct correspondence between the labels generated by the individual clusterings leads to the need for this mapping component. For instance, the co-association (or co-occurrence) matrix [2] is an example of a representation generated by an ensemble mapping that side-steps the label correspondence problem, at a computational cost of $O(N^2)$. The maximum likelihood mapping [8] is another example of ensemble mapping in which the re-labelling problem is formulated as a weighted bipartite matching problem and is solved using the Hungarian method [9] with a computational cost of $O(k^3)$ where k is the number of clusters.

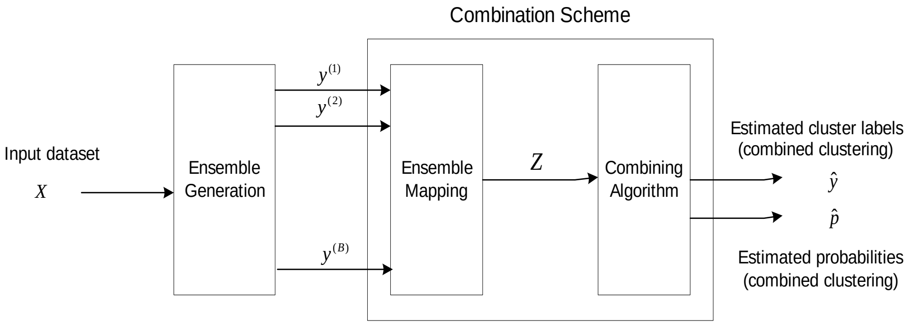


Fig. 1. Schematic model of cluster ensembles

The second sub-element of a combining scheme is the combining algorithm which uses $\mathbf{Z}$ to generate the combined clustering $\hat{\mathbf{y}}$. A potential derivative of the cluster ensemble is the estimation of the probabilities $\hat{p}$ with which data points belong to the combined clusters. The combining algorithm often lends itself to a clustering problem, where the data is given by the new representation $\mathbf{Z}$. It is noted that if the label correspondence problem is resolved and the number of clusters c in the base clusterings $\{\mathbf{y}^{(i)}\}_{i=1}^{B}$ is the same as the number of clusters k in the combined clustering $\hat{\mathbf{y}}$, majority voting [6] or maximum likelihood classification [8] can be readily applied. However, if $c \neq k$, co-association-based consensus functions are often applied [2–4]. While allowing arbitrary cluster structures to be discovered, co-association-based consensus functions are computationally expensive and hence not practical for large datasets.

Re-sampling methods are well established approaches for estimating improved data statistics [10]. In particular, bagging [11] has been introduced in regression and classification. In bagging, the training dataset of size N is perturbed using bootstrap re-sampling to generate learning datasets by randomly sampling N patterns with replacement. This yields duplicate patterns in a bootstrap dataset. The bootstrap re-sampling process is independently repeated B times and the B datasets are treated as independent learning sets.

Dudoit and Fridlyand [6] used bagging with the Partitioning Around Medoids (PAM) clustering method to improve the accuracy of clustering. They use two methods for combining multiple partitions. The first applies voting and the second creates a new dissimilarity matrix similar to the co-association matrix used in [2]. In the voting method, the same number of clusters is used for clustering and combining, and the input dataset is clustered once to create a reference clustering. The cluster labels of each bootstrap replication are permuted such that they fit best to the reference clustering. They reported that the bagged clustering were generally as accurate and often significantly more accurate than a single clustering. Fischer and Buhmann [8] applied bagging to improve the quality of the path-based clustering method. They critiqued the use of a reference clustering in the mapping method of Dudoit and Fridlyand [6], arguing that it imposes undesirable influence. Instead, they selected a re-labelling out of all $k!$ permutations for a clustering, such that it maximizes the sum over the empirical cluster assignment probabilities estimated from previous mappings, over all objects of the new mapping configuration. The problem of finding the best permutation is formulated as a weighted bipartite matching problem and the Hungarian method is used to solve a maximum bipartite matching problem. They reported that bagging increases the reliability of the results and provides a measure of uncertainty of the cluster assignment. Again, in this method, the number of clusters used in the ensemble is the same as the number of combined clusters. Minaei, Topchy and Punch [7] empirically investigated the effectiveness of bootstrapping with several consensus functions by examining the accuracy of the combined clustering for varied resolution of partitions (i.e., number of clusters) and ensemble size. They report that clustering of bootstrapping leads to improved consensus clustering of the data. They further conclude that the the best consensus function remains an open question, as different consensus functions seem to suit different cluster structures.

In this paper, we propose an ensemble mapping representation based on the generated clusters, as high-level data granules. Re-labelling of clusters is based on maximizing individual cluster similarity to incrementally-accumulated clusters. Based on this representation, different combining algorithms can be used such as hierarchical clustering algorithms, for instance. Here, group average (i.e. average link) hierarchical meta-clustering is applied. We experimentally investigate the effectiveness of the proposed consensus function, with bootstrap re-sampling, and the k-means as the underlying clustering algorithm.

2 Cluster-Based Cumulative Ensemble

2.1 Ensemble Mapping

The ensemble representation consists of a cumulative $c \times N$ matrix $\mathbf{Z}$ summarising the ensemble outputs, where c is a given number of clusters that is used in generating multiple clusterings, such that $k \leq c \ll N$ where k is the number of combined clusters. The data values in $\mathbf{Z}$ reflect the frequency of occurrence of each pattern in each of the accumulated clusters.

The k-means algorithm with the Euclidean distance is used to generate a clustering $\mathbf{y}^{(b)} = \pi(\mathbf{X}^{(\mathbf{b})}, c)$ of a bootstrapped learning set in $\{\mathbf{X}^{(b)}\}_{b=1}^{B}$, where B is the size of the ensemble, and $\mathbf{y}^{(b)}$ is an N-dimensional labeling vector. That is, π is a mapping function $\pi : \mathbf{X}^{(b)} \rightarrow \{0, \cdots, c\}$, where '0' label is assigned to patterns that didn't appear in the bootstrap learning set $\mathbf{X}^{(b)}$.

Each instance of the $c \times N$ matrix, denoted by $\mathbf{Z}^{(b)}$, is incrementally updated from the ensemble $\{\mathbf{y}^{(b)}\}_{b=1}^{B}$ as follows.

1. $\mathbf{Z}^{(1)}$ is initialized using $\mathbf{y}^{(1)}$, as given below. Re-labelling and accumulation start by processing clustering $\mathbf{y}^{(2)}$.

$$z_{ij}^{(1)} = \begin{cases} 1 & \text{if object } j \text{ is in cluster } i \text{ according to clustering } \mathbf{y}^{(1)} \\ 0 & \text{otherwise} \end{cases}$$

2. Let each cluster in a given clustering $\mathbf{y}^{(b+1)}$ be represented by a binary N-dimensional vector $\mathbf{v}$ with 1's in entries corresponding to the cluster members and 0's otherwise. Let each cluster extracted from the rows $\mathbf{z}_i^{(b)}$ of $\mathbf{Z}^{(b)}$ be represented by the binary N-dimensional vector $\mathbf{w}$ whose entries are 1's for non-zero columns of $\mathbf{z}_i^{(b)}$ and 0's otherwise. Compute the similarity between each pair of vectors $\mathbf{v}$ and $\mathbf{w}$ using the Jaccard measure given as $J(\mathbf{v}, \mathbf{w}) = \mathbf{vw}/(\|\mathbf{v}\|^{\mathbf{2}} + \|\mathbf{w}\|^{\mathbf{2}} - \mathbf{vw})$
3. Map each cluster label $i \in \{1, \cdots, c\}$ in clustering $\mathbf{y}^{(b+1)}$ to its most similar cluster labelled $j \in \{1, \cdots, c\}$ of the previously accumulated clusters represented by the rows of $\mathbf{Z}^{(b)}$. Hence, increment the entries of row j of $\mathbf{Z}^{(b)}$ corresponding to members of the cluster labelled i in clustering $\mathbf{y}^{(b+1)}$.
4. $\mathbf{Z}^{(b+1)} \leftarrow \mathbf{Z}^{(b)}$. The mapping process is repeated until $\mathbf{Z}^{(B)}$ is computed.

The cumulative cluster-based mapping of the ensemble culminates in the matrix $\mathbf{Z} = \mathbf{Z}^{(B)}$, as a voting structure that summarises the ensemble. While in the maximum likelihood mapping [8], the best cluster label permutation is found and $c = k$ is used, in this paper, each cluster is re-labelled to match its most similar cluster from the accumulated clusters. This is done for the following reasons. First, since the base clusterings represent high resolution partitions of non-identical bootstrap learning sets, this leads to highly diverse clusterings, such that finding the best permutation becomes less meaningful. For quantitative measures of diversity in cluster ensembles, the reader is referred to [5]. Second, since the accumulated clusters will be merged in a later stage by the combining algorithm, we are most concerned at this stage in a mapping which maximizes the similarities and hence minimizes the variance of the mapped clusters.

We found that this matching method can occasionally result in a cumulative cluster to become singled out when no subsequently added clusters are mapped to it. If a hierarchical clustering algorithm is used, this problem can lead to a degenerate dendrogram and empty cluster(s) in the combined clustering. Therefore, we detect this condition, and the corresponding solution is discarded. Usually, a good solution is reached in a few iterations. An alternative remedy is to match each of the cumulative clusters to its most similar cluster from each subsequently mapped clustering, instead of the reverse way. This ensures that the above mentioned condition does not occur, but it can introduce influence from earlier clusterings and less incorporation of the diversity in the ensemble.

An advantage of this representation is that it allows several alternative views (interpretations) to be considered by the combining algorithm. For instance, $\mathbf{Z}$ may be treated as a pattern matrix. This allows different distance/similarity measures and combining algorithms to be applied to generate the combined clustering. Alternatively, $\mathbf{Z}$ may be treated as the joint probability distribution of two discrete random variables indexing the rows and columns of $\mathbf{Z}$. This allows for information theoretic formulations for finding of the combined clusters.

Furthermore, the size of this representation is $c \times N$ versus N^2 for the co-association-based representation, where $c \ll N$. While, in the case of the co-association matrix, the hierarchical clustering algorithm runs on the $N \times N$ matrix, in the case of the cluster-based cumulative representation, it runs on a $c \times c$ distance matrix computed from the $c \times N$ matrix $\mathbf{Z}$.

2.2 Combining Using Hierarchical Group Average Meta-clustering

Motivated by what is believed to be a reasonable discriminating strategy based on the average of a chosen distance measure between clusters, the proposed algorithm is the group average hierarchical clustering. The combining algorithm starts by computing the distances between the rows of $\mathbf{Z}$ (i.e. the cumulative clusters). This is a total of $\binom{c}{2}$ distances, and one minus the binary Jaccard measure, given in Section 2.1, is used to compute the distances. The group-average hierarchical clustering is used to cluster the clusters, hence the name meta-clustering. In this algorithm, the distance between a pair of clusters $d(C_1, C_2)$ is defined as the average distance between the objects in each cluster, where the objects in this case are the cumulative clusters. It is computed as follows, $d(C_1, C_2) = mean_{(\mathbf{z}_1,\mathbf{z}_2)\in C_1 \times C_2} d(\mathbf{z}_1, \mathbf{z}_2)$, where $d(\mathbf{z}_1, \mathbf{z}_2) = 1 - J(\mathbf{z}_1, \mathbf{z}_2)$

The dendrogram is cut to generate k meta-clusters $\{M_j\}_{j=1}^k$ representing a partitioning of the cumulative clusters $\{\mathbf{z}_i\}_{i=1}^c$. The merged clusters are averaged in a $k \times N$ matrix $\mathbf{M} = \{m_{ji}\}$ for $j \in \{1, \cdots, k\}$ and $i \in \{1, \cdots, N\}$. So far, only the binary version of the cumulative matrix has been used for distance computations. Now, in determining the final clustering, the frequency values accumulated in $\mathbf{Z}$ are averaged in the meta-cluster matrix $\mathbf{M}$ and used to compute the cluster assignment probabilities. Then, each object is assigned to its most likely meta-cluster. Let M be a random variable indexing the meta-clusters and taking values in $\{1, \cdots, k\}$, let X be a random variable indexing the patterns and taking values in $\{1, \cdots, N\}$, and let $\hat{p}(M = j|X = i)$ be the conditional probability of each of the k meta-clusters, given an object i, which we also write as $p(M_j|x_i)$. Here, we use x_i to denote the object index of the pattern $\mathbf{x}^{(i)}$, and we use M_j to denote a meta-cluster represented by the row j in $\mathbf{M}$. The probability estimates $\hat{p}(M_j|x_i)$ are computed as $\hat{p}(M_j|x_i) = \frac{m_{ji}}{\sum_{l=1}^k m_{li}}$.

3 Experimental Analysis

Performance is evaluated based on the error rates which are computed by solving the correspondence problem between the labels of a clustering solution and the true clustering using the Hungarian method.

3.1 Experiments with Artificial Data

The artificial datasets are shown in Figure 2. The first, called “Elongated-ellipses” consists of 1000 2D points in 2 equal clusters. The “Crescents” dataset consists of 1000 2D points in 2 equal clusters. The “Differing-ellipses” consists of 250 2D points in 2 clusters of sizes 50 and 200. The dataset called “8D5K” was generated and used in [1]. It consists of 1000 points from 8D Gaussian distributions (200 points each). For visualization, the “8D5K” data is projected onto the first two principal components.

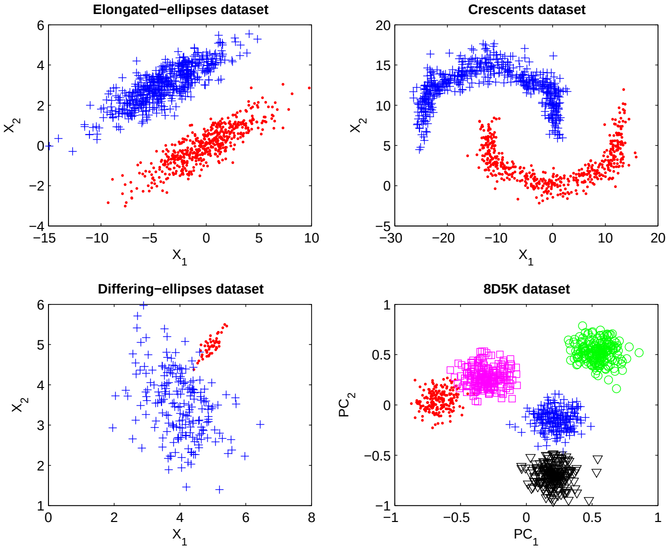


Fig. 2. Scatter plots of the artificial datasets. The last 8 dimensional dataset is projected on the first 2 principal components

For each dataset, we use $B = 100$, and vary c. We measure the error rate of the corresponding bagged ensemble at the true number of clusters k and compare it to the k-means at the same k. The results in Figure 3 show that the proposed bagging ensemble significantly lowers the error rate for varied cluster structures. In order to illustrate the cluster-based cumulative ensemble, we show in Figure 4 (a) a plot of the points frequencies in each of the accumulated clusters at $c = 4$ for the “elongated-ellipses” dataset. The points are ordered such that the first 500 points belong to the first cluster followed by 500 from the second cluster. The dendrogram corresponding to the hierarchical group average meta-clustering on the 4 cumulative clusters is shown in Figure 4 (b).

3.2 Experiments with Real Data

We use six datasets from the UCI machine learning repository. Since the Euclidean distance is not scale invariant, we standardize the features for those datasets in which the scales widely vary for the different features. The datasets used are, (1) the iris plant dataset, (2) the wine recognition dataset, (3) the Wisconsin

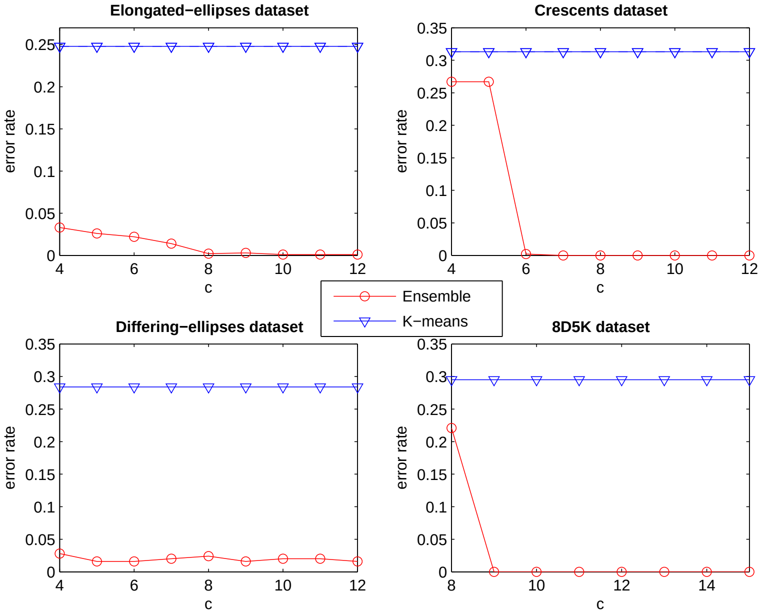
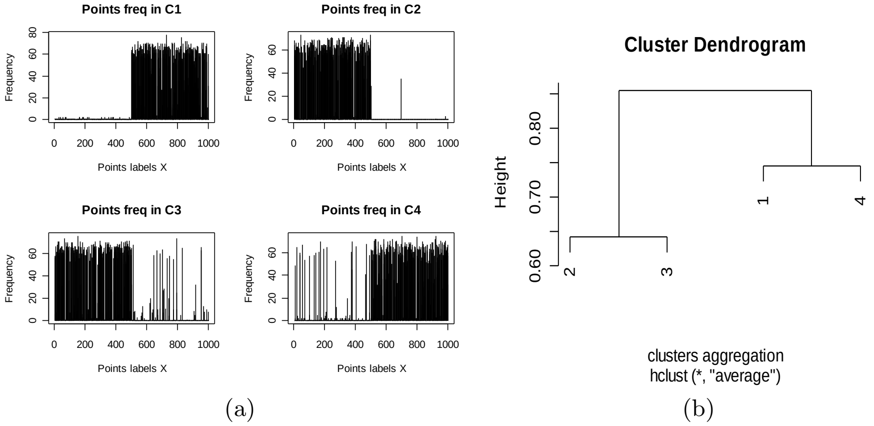


Fig. 3. Error rates for artificial datasets using the bagged cluster ensembles and the k-means algorithm at given k



(a) (b)

Fig. 4. (a) Accumulated clusters. (b) Generated dendrogram.

breast cancer dataset, (4) the Wisconsin diagnostic breast cancer (WDBC), (5) a random sample of size 500 from the optical recognition of handwritten digits dataset, and (6) a random sample of size 500 from the pen-based recognition of handwritten digits dataset. We standardized the features for the wine recognition and the WDBC datasets. The mean error rates of the k-means (over 100 runs), at the true k, for the above datasets are, 0.2007, 0.0378, 0.0395, 0.0923, 0.2808, 0.3298, respectively.

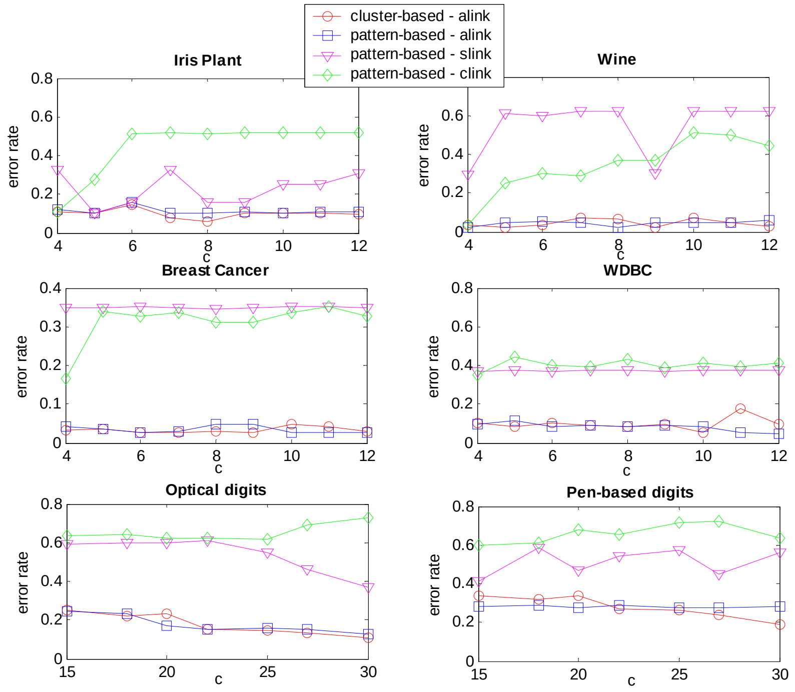


Fig. 5. Error rates on the real datasets for the proposed ensemble versus co-associations-based ensembles using single, complete and average link.

Figure 5 shows a comparison of the cluster-based cumulative ensembles with hierarchical group average (denoted in Figure 5 by cluster-based alink) to pattern co-association-based ensembles, when single, complete and average link variants of the hierarchical clustering are applied (denoted by pattern-based slink, clink, and alink, respectively). In the experiments, we use $B = 100$ and k corresponding to the true number of clusters. The results show that the cluster-based alink ensembles perform competitively well compared to pattern-based alink ensembles. On the other hand, the co-association-based single and complete link ensembles showed poor performance.

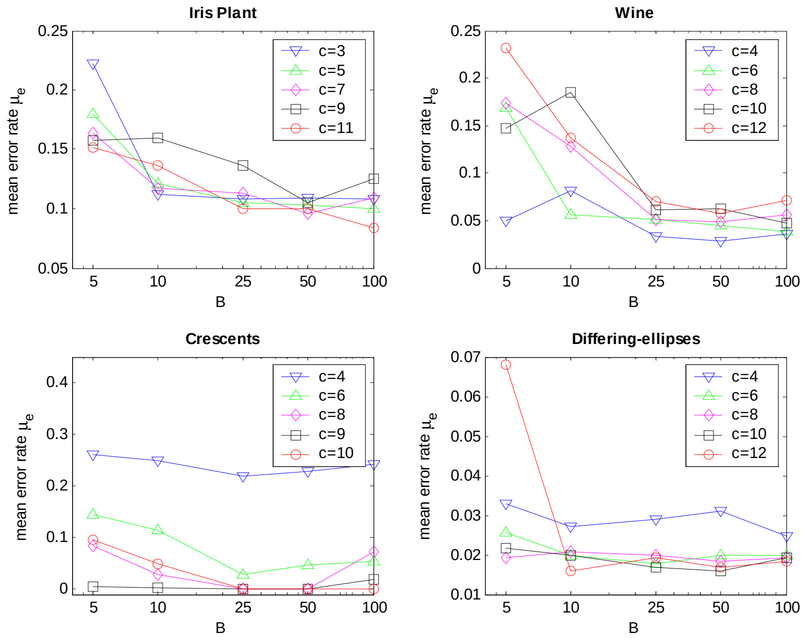


Fig. 6. Effect of ensemble size B. The X-axis is log scale.

3.3 Varying The Ensemble Size

We study the effect of the ensemble size B, for values of $B \leq 100$. Figure 6 shows the mean error rates on real and artificial datasets for $B = 5, 10, 25, 50,$ and 100, and for varying number of base clusters c. Each ensemble at a given c and B is repeated $r = 5$ times and the mean is computed. There is a general trend of reduction in error rates as B increases. However, we observe that most gain in accuracy occurs for $B = 25$, and 50. We also observe that the differences between the error rates of ensembles of varying values of c tend to decrease as B increases, i.e., the variability of the error rates corresponding to different values of c is reduced when B is increased. However, in some cases, it is noted that that amount of reduction in the error depends on c. For instance, this can be observed for $c = 4$ in the crescents and differing-ellipses datasets.

4 Conclusion

The proposed cluster-based cumulative representation is more compact than the co-association matrix. Experimental results on artificial datasets emphasised the potential of the proposed ensemble method in substantially lowering the error rate, and in finding arbitrary cluster structures. For the real datasets, the cluster-based cumulative ensembles, using group average hierarchical clustering, significantly outperformed co-association-based ensembles, using the single and complete link algorithms. They showed competitive performance compared to co-association-based ensembles, using the group average algorithm. In [12], the group average algorithm is shown to approximately minimize the maximum cluster variance. Such model seems to represent a better fit to the data summarised in **Z**. A further potential benefit of this paper is that co-association-based consensus functions other than hierarchical methods, such as [3, 4], can also be adapted to the cluster-based cumulative representation, rendering them more efficient. This will be investigated in future work.

Acknowledgments

This work was partially funded by an NSERC strategic grant.

References

1. A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal on Machine Learning Research (JMLR)*, 3:583–617, December 2002.
2. A. Fred and A.K. Jain. Data clustering using evidence accumulation. In *Proceedings of the 16th International Conference on Pattern Recognition. ICPR 2002*, volume 4, pages 276–280, Quebec City, Quebec, Canada, August 2002.
3. H. Ayad and M. Kamel. Finding natural clusters using multi-clusterer combiner based on shared nearest neighbors. In *Multiple Classifier Systems: Fourth International Workshop, MCS 2003, UK, Proceedings.*, pages 166–175, 2003.
4. H. Ayad, O. Basir, and M. Kamel. A probabilistic model using information theoretic measures for cluster ensembles. In *Multiple Classifier Systems: Fifth International Workshop, MCS 2004, Cagliari, Italy, Proceedings*, pages 144–153, 2004.
5. L. I. Kuncheva and S.T. Hadjitodorov. Using diversity in cluster ensembles. In *IEEE International Conference on Systems, Man and Cybernetics, Proceedings*, The Hague, The Netherlands., 2004.
6. S. Dudoit and J. Fridlyand. Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19(9):1090–1099, 2003.
7. B. Minaei, A. Topchy, and W. Punch. Ensembles of partitions via data resampling. In *Intl. Conf. on Information Technology: Coding and Computing, ITCC04, Proceedings*, Las Vegas, April 2004.
8. B. Fischer and J.M. Buhmann. Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11):1411–1415, 2003.
9. H. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97, 1955.
10. R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. John Wiley & Sons, 2001.
11. Leo Breiman. Bagging predictors. *Machine Learning*, 26(2):123–140, 1996.
12. S. Kamvar, D. Klein, and C. Manning. Interpreting and extending classical agglomerative clustering algorithms using a model-based approach. In *Proceedings of the 19th Int. Conf. Machine Learning*, pages 283–290, 2002.