A Study of Consensus Function in Cluster Ensemble

Ms. Urvashi Soni[1], Dr. Sunita Dwivedi[2]
1 Assistant professor, SAGE University Indore, urvashikushsoni2016@gmail.com
2 Associate Professor, MCRPV Bhopal, ddwivedi2001@gmail.com

ABSTRACT

The important phase in ensemble clustering is the consensus function. In terms of what is the goal for comparison in the consensus process, this study divides all consensus functions into four categories: partition-partition (P-P) comparison, cluster-cluster (C-C) comparison, member-in-cluster (MIC) voting, and member-member (M-M) co-occurrence. Each ensemble clustering approach is divided into two steps: generation and consensus. P-P comparison approaches, also known as median partition approaches, aim to solve an optimization problem by maximizing the total similarity to the specified partitions. C-C Comparison treats a cluster as a unit and examines cluster similarity. MIC Voting is a concept shared by several systems that focus on the relationship among members and their clusters across all partitions. The M-M co-occurrence method transforms the consensus partitioning problem into a co-association matrix partitioned problem.

Keywords: Consensus Function; Clustering; Voting; Co-Occurrence; Ensemble
DOINumber:10.48047/nq.2022.20.22.NQ10269 **NeuroQuantology2022;20(22):2768-2774**

1. INTRODUCTION

As is well known, no clustering algorithm can correctly determine the basic clustering structure for all datasets. Naturally, this raises the question: how can we determine which clustering technique to apply on a given unknown dataset? Furthermore, for the same dataset, some stochastic clustering algorithms may produce different clustering outcomes with different initialization values/ parameters. The next question is, which clustering result should we believe? In clustering analysis, one approach to the problem is to employ numerical clustering validation techniques, which evaluate the validity of clustering findings based on a variety of parameters. As it is also true that no cluster validation algorithm has been claimed to be unbiased about the results.

Combining various clustering results, often referred to as ensemble clustering, consensus clustering, or cluster aggregation, is an alternative response to the above query. The concept of ensemble clustering formalizes the notion that highlighting the shared organization among several clustering results would be achieved by aggregating them into a single representative or consensus result. One of the expected characteristics of ensemble clustering is that its results must be more reliable, new, and stable than those of a single clustering algorithm. The inherent structure of the dataset or the ground truth may not be the optimum outcome; hence it is not necessary for ensemble clustering to have these features. Only one fact may be assured: the consensus of all clustering algorithms may substitute for probable faults by individual algorithms, and the total clustering result is more statistically trustworthy than any single one.

The definition of a suitable consensus function poses a significant difficulty to ensemble clustering. In terms of the consensus function, ensemble clustering algorithms are typically divided into two broad categories: the item co-occurrence method and the median partition approach. The crucial stage in ensemble clustering is the consensus function. In terms of the comparison targets used in the consensus

process, we group all consensus functions into four categories: member-in-cluster (MIC) voting, partition-partition (P-P) comparison, cluster-cluster (C-C) comparison, and member-member (M-M) co-occurrence. Instead of listing every ensemble clustering algorithm, we mainly concentrate on the fundamental ideas in each class.

2. Consensus Functions

2.1. P–P Comparison

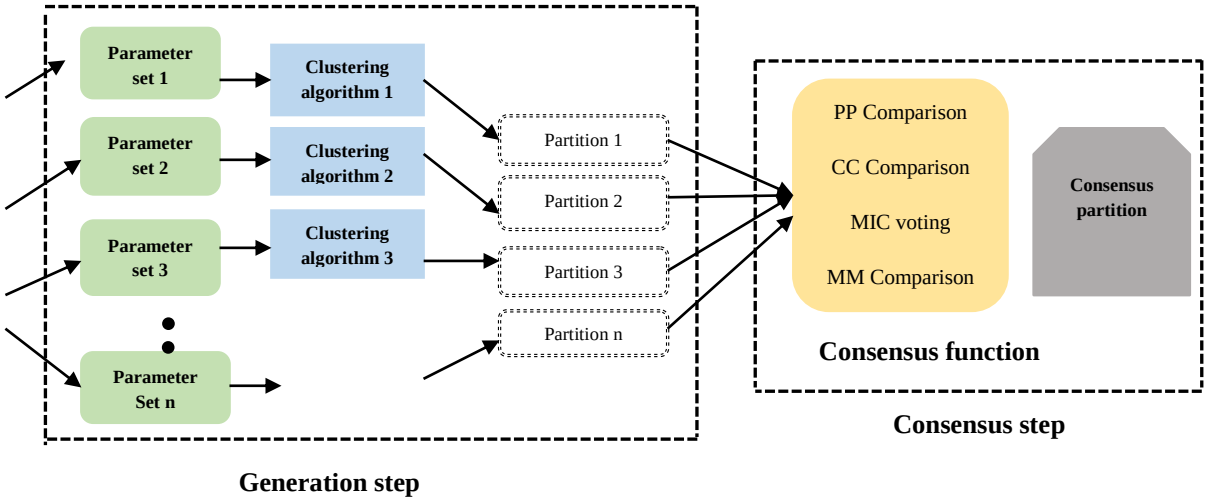


Figure 1 Diagram of a general process of ensemble clustering. The consensus step may have four different types of consensus functions, namely P–P comparison, C–C comparison, MIC voting, M–MCO-OCCurrence

P-P Comparison approaches, also known as median partition approaches, aim to solve an optimization problem by maximizing the total similarity to the provided partitions. Therefore, the P-P similarity/dissimilarity comparison is a crucial component of this class of consensus functions. Optimization problem can be written as: $\boldsymbol{Z^x = argmax_{P\in P_x} \sum_{j=1}^{R} \Gamma(Z, Z_j)}$

Γ is a similarity measure. The partition with the greatest degree of similarity to every other partition in the ensemble clustering is known as the consensus partition. The key theoretical conclusions have been reached in the specific scenario where the symmetric clustering algorithm exists. The issue becomes the minimization of the total dissimilarity concerning the cluster ensemble whenever dissimilarity between partitions. The P-P comparison can make use of a wide variety of similarity and dissimilarity measurements between partitions. These measurements, such as the Rand index (RI) [16], the Adjusted Rand index (ARI), the normalized mutual information (NMI) [6], and the Jaccard index (JI), are widely used in the research on the external clustering validity index [16]. In data clustering, the Rand index or Rand measure is a measure of the similarity between two data clustering. The adjusted Rand index is a variant of the Rand index that takes into account the random grouping of items. The Rand index can be used to examine how closely two clustering techniques produce results. Where the number of times a pair of components from two different clustering algorithms are members of the same cluster. To establish whether the two cluster findings are similar to one another, the Adjusted Rand score is presented. The "RI" in the calculation stands for the Rand index, which compares two cluster results by taking into account all points found in the same cluster. When we are given the cluster labels, normalized mutual information (NMI) provides us with a reduction in the entropy of class labels. When we know the cluster labels, NMI sort of informs us how much the ambiguity about class labels diminishes. It is comparable to how decision trees gain knowledge. The Jaccard Index calculates the frequency of joining pairs of items in two clustering data sets and the frequency of seeing pairs exclusively in one set in the case of partitioning findings.Along with the external validity indices mentioned above, the Mirkin distance is a well-liked indicator of partition similarity [14]. Mirkin distance thus is defined as $\boldsymbol{M(Z_1, Z_2) = n_{01} + n_{10}}$ (symmetric distance representing the number of disagreement between two partitions)

$$M_{uv}(Z_1, Z_2) = \begin{cases} 1 \; if\big(Z_1(u)\big) = Z_1(v) \; and \; \big(Z_1(u)\big) \neq Z_1(v)) \\ 0 \; or \; \big(Z_1(u)\big) \neq Z_1(v) \; and \; \big(Z_1(u)\big) = Z_1(v)) \end{cases}$$

Otherwise $M(Z_1, Z_2) = \sum_{u,v} M_{uv(Z_1,Z_2)}$

2.2. C– C Comparison

C-C Comparison evaluates a cluster as a whole and assesses how similar different clusters are to one another. In other words, clustering clusters provide the foundation of C-C comparison algorithms. Meta-clustering algorithm (MCLA), a common C-C comparison algorithm, was proposed by Strehl and Ghosh [20]. Each cluster in MCLA has a hyperedge to symbolize it. Each data object is assigned to the collapsed hyperedge in which it participates most actively by MCLA, which clusters and collapses similar hyperedges. A graph-based clustering determines which hyperedges are linked for the sake of collapsing. A meta-cluster is used to describe each cluster of hyper-edges.

a. **Construct meta-graph** - The term "meta-graph" refers to a regular undirected graph that has all of the r indicator vectors h as vertices. The edge weights are related to how similar the vertices are to one another. The binary Jaccard measure was used in MCLA by Strehl and Ghosh. The meta-graph is r-partite since there are no edges between the vertices of the same partition because the clusters do not overlap in individual partitions.
b. **Hyper edge** groups - This stage involves obtaining matching labels by dividing the meta-graph into k-balanced meta-clusters. In this stage, many graph-partitioning algorithms may be used. The indicator vectors h were clustered into k meta-clusters by Strehl and Ghosh [20] using the METIS program. Because each vertex in the meta-graph has a different cluster label, a meta-cluster represents a collection of corresponding labels.
c. **Collapse Meta-clusters** - The hyperedges are combined into a single meta-hyper edge for each of the k meta-clusters. A description of each object's level of correlation with the relevant meta-cluster is given in the entry for each object in the association vector that each meta-hyper edge holds. A certain meta-indicator cluster's vectors h are averaged to determine the level.
d. **Compete for Objects** - The highest entry in the association vector is used to determine which meta-cluster each object is assigned.

The experiment results in Strehl and Ghosh [19] demonstrated that when compared to the cluster-based similarity partition algorithm (CSPA) and the hypergraph-partitioning algorithm (HGPA). MCLA should be the most suitable in terms of time complexity and quality.

2.3. MICVoting

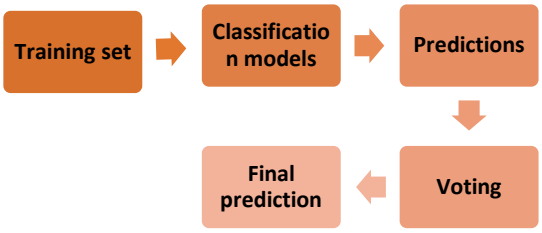
A common concept in approaches is MIC Voting, which concentrates on the relationship between members and their clusters across all partitions. A voting classifier is a model that gains training data from a large ensemble of models and forecasts an output (class) based on the class with the highest likelihood of being the output. In general, boosting will strive to generate strong models that are less biased than their components, whereas bagging will primarily focus on creating an ensemble model with less variance than its components (even if variance can also be reduced).The winning clusters have a strong likelihood of being included in the final consensus partition through voting by data items.

a. **RelabelingandVoting -**The voting-merging (VM) algorithm was proposed [7]. VM consists of three procedures, namely the partition procedure, the voting procedure, and finally the merging procedure. The critical two procedures of VM are the voting and merging procedures.

A merging procedure merges clusters that are closest to each other. If a data point is assigned to both clusters which have been mapped, one vote is issued to the common cluster. The same action is done to the remaining clusters iteratively until all clusters in partitions are mapped (relabeled). After relabeling, the voting procedure takes place.

As a solution to the issue of aligning clustering labels produced with various numbers of clusters, [3] suggested the concept of cumulative voting. The cumulative clustering algorithm, in contrast to previous voting techniques, computes a probabilistic mapping. Virtual cumulative voting, also known as weighted voting, transforms an input k_i-partition into a k_0-partition with cluster labels that match the labels

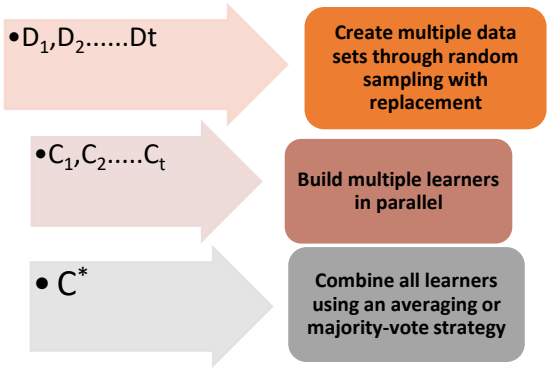
of the reference clusters. [2]Specified the smallest average squared distance between the first summary of the ensemble. The selection criterion for the reference clustering is



mapped partitions and the ideal representation of the ensemble as a criterion for obtaining a

b. Bagging - Bagging also known as Bootstrap aggregating, is a resampling scheme, which is designed to improve the quality of clustering. In BagClust1, the clustering procedure is repeatedly applied to each bootstrap sample and the final partition is obtained by plurality voting; that is, by taking the majority cluster for each observation. Unlike other voting methods, a probabilistic mapping is computed in the cumulative clustering algorithm. Cumulative voting virtually is sometimes referred to as weighted voting. A criterion was defined [3] for obtaining a first summary of the ensemble. The selection criterion is formulated for the reference clustering.In the bagging technique, each model is given identical weight, however, in the boosting technique, the new models are weighted according to their performance. Observations that the prior model incorrectly identified are included in new subsets of data that are used for training in boosting. Training data subsets are produced at random for bagging. Without the aid of the original patterns in X, [21] and colleagues suggested a finite mixture model to address the consensus problem using only the label information provided by the contributing clustering techniques [22]. The primary premise is that the developed based on the maximal information content.

labels are modeled as random variables selected from a probability distribution that is characterized as a combination of multivariate component densities. The goal of consensus clustering is currently defined as a problem of maximum-likelihood estimation. By maximizing the log-likelihood function concerning the unknown parameters, the best mixture density for dataset Y can be found. As a result, the original clustering issue in the space of data X has been changed into a space of fresh multivariate features Y with the aid of numerous clustering techniques. The final objective is to discretely label the data X using a deceptive method of density estimation of Y. The expectation-maximization (EM) algorithm can be used to solve the maximum likelihood problem. The full set of data (Y, Z) must exist to use the EM method. Knowing the value of z_n makes it simple to identify the element that produces the data. The Bagging approach has the benefit of managing higher dimension data sets very effectively. It maintains accuracy for missing data and controls missing quantity. The Bagging approach has drawbacks. The final prediction won't provide an exact value for the regression model because it depends on the mean predictions from subgroup trees.



c. Boosting- A set of predictors is built via boosting. According to this method, early learners first fit simple models to the data before looking for mistakes in the data. The random sample is fit, and each step aims to increase accuracy over the previous one. An input's weight is increased when a hypothesis incorrectly classifies it, increasing the likelihood that the subsequent hypothesis will classify it properly. Weak learners are transformed into better-performing models through this approach. Advantages: supports several loss functions (example, "binary: logistic") and functions well in interactions. Disadvantages: inclined to overfit. Requires meticulous adjustment of several hyper-parameters.

Is bagging therefore preferable to boosting? So, there is no obvious victor. According to the information and problem stated. To achieve satisfactory results, a trade-off between variance and bias must be found. There is a big variance that needs to be addressed if the model performs well on training data but poorly on test data. Similarly to this, a model cannot be used for generalization if it is not learning enough from training data. Propose experimenting with the dataset to observe how changing the parameters affects the test data score. Additionally, you can experiment with various ensembles, such as XGBoost.

2.4.M–MCo-occurrence

The consensus partition problem is changed into a co-association matrix partitioning problem using the M-M co-occurrence technique. The frequency of two data objects' Co-Occurrence in all partitions, or their likelihood of running into one another, is the entry of the co-association matrix. The two data objects co-occurrence in all partitions expressed as:

$$A_{ij} = \frac{1}{R}\sum_{r-1}^{R} \delta\left(Z_r(x_i), Z_r(x_j)\right)$$

Where Z_r (x_i) represents the associated cluster label of data object x_i in partition P_r and δ (a, b) is 1 if a=b & otherwise 0.

a. Tree-basedMethods - The idea of evidence accumulation for combining the results of multiple clustering [10]. A measure of similarity between patterns, which benefits the combination of the clustering results, was proposed. The R partitions of N data points were mapped into a $N * N$ co-association matrix. Different clustering algorithms could be applied to the similarity matrix. $C_{(i,j)=\frac{n_{ij}}{R}}$

Where n_{ij} is the number of times among the R partitions that the pattern pair (i,j) is given the same cluster. The similarity matrix could be subjected to various clustering techniques. To extract the combined data partition, [10] investigated the evidence-accumulation clustering method using the single-link and average-link hierarchical agglomerative algorithms. Based on the idea of mutual information and variance analysis using bootstrapping, they also proposed a theoretical framework and optimality criteria for the examination of clustering combination outcomes.

b. Graph-basedMethods - A graph-partitioning technique produced the co-association matrix. The similarity matrix entry for two items that belong to the same cluster is given a one by CSPA. Assume mathematically that the rth partition's binary membership indicator matrix is $H_{(R)}$METIS, which was used [20] becauseofitsrobustandscalableproperties. Co-

association matrix is given by equation $\boldsymbol{A} = \frac{\mathbf{1}}{\boldsymbol{R}}\boldsymbol{H}\boldsymbol{H}^{\boldsymbol{T}}$.

c. **ResamplingMethod -** The clustering algorithm of choice can be applied to perturbed (Flustered) datasets, and the consensus among multiple runs can be assessed. Assuming a resampling scheme and a clustering algorithm had been selected, devised a method for representing and quantifying the agreement among the clustering runs over the perturbed datasets.Developed a consensus clustering method in conjunction with resampling techniques [15]. It provided the consensus across multiple runs of a clustering algorithm and the ability to assess the stability of the discovered clusters.

Connecting matrices of all the perturbed datasets $\boldsymbol{A}_{(\boldsymbol{i},\boldsymbol{j})} = \frac{\sum_{\boldsymbol{r}} \boldsymbol{A}^{(\boldsymbol{r})}(\boldsymbol{i},\boldsymbol{j})}{\sum_{\boldsymbol{r}} \boldsymbol{I}^{(\boldsymbol{r})}(\boldsymbol{i},\boldsymbol{j})}$

That is, the number of times items I and j are allocated to the same cluster divided by the total number of times both items are selected is recorded in the entry of $A_{i\,j}$. Then, A is subjected to an agglomerative hierarchical tree construction technique to produce a dendrogram of item adjacencies. Based on the consensus matrix, summary statistics accounting for the stability of a specific cluster as well as of a cluster's members were defined. These statistics can be used to rank the clusters according to their stability and to determine which items within each cluster are more pertinent.

Cluster consensus defined as $-\boldsymbol{m}(\boldsymbol{k}) = \frac{\mathbf{1}}{\boldsymbol{N_k}(\boldsymbol{N_k}-\mathbf{1})/\mathbf{2}} \sum_{\substack{\boldsymbol{i},\boldsymbol{J}\in\boldsymbol{I_k} \le \boldsymbol{m} \\ \boldsymbol{I}<\boldsymbol{J}}} \boldsymbol{A}(\boldsymbol{i},\boldsymbol{j})$

Consensus item $\boldsymbol{m_j}(\boldsymbol{k}) = \frac{\mathbf{1}}{\boldsymbol{N_k}-\mathbf{1}\{\boldsymbol{e_i}\in\boldsymbol{I_K}\}} \sum_{\substack{\boldsymbol{i}\in\boldsymbol{I}_{\boldsymbol{k}} \\ \boldsymbol{i}\neq\boldsymbol{j}}} \boldsymbol{A}(\boldsymbol{i},\boldsymbol{j})$

Where the condition is an indicator function that, when true, equals 1 and, when false, 0; additionally, a [0, 1] range-defined empirical cumulative distribution function (ECDF).

3. Conclusion

There is no guaranteed optimal algorithm for the clustering problem due to its unsupervised nature. Consensus clustering has gained a lot of attention within the last decade, primarily because of the following reasons-First, the clustering problem's unsupervised nature prevents there from being a guaranteed optimal algorithm; Second, Different clustering techniques can be applied to the same data under different circumstances, i.e. when various initialization parameters are used, or even when the same parameters are applied over several runs.; Lastly, merging the clustering outcomes of the same collection of data points across many datasets is another intriguing but difficult subject.

The core ideas of consensus clustering, which, in terms of consensus functions, can be divided into four groups: P-P comparison, C-C comparison, MIC voting, and M-M co-occurrence. The goal of P-P comparison-type consensus functions is to maximize the average similarity between the input partitions and the best consensus partitions. The distance metric can be the Mirkin distance or any information-theoretic distance metric. A meta-graph, whose vertices are clusters and weighted links are similarities of pairwise clusters, is created by C-C comparison-type consensus functions that take into account the similarity or dissimilarity between a pair of clusters from various clustering results. The meta-graph is then clustered using graph-based clustering techniques. When using MIC voting-type consensus functions, each data object in each cluster is given a vote, and the votes are totaled across all clustering outcomes. By using M-M co-occurrence-type consensus functions, the remaining clustering issues can be resolved using either tree-based algorithms, such as hierarchical clustering algorithms, or graph- or hypergraph-based algorithms. These algorithms generate an association matrix in terms of the frequency of co-occurrence of every pair of data points in all clustering results.

References

1. Abu-Jamous, B., Fa, R., Roberts, D.J. and Nandi, A.K. (2013). The paradigm of tunable clustering using binarization of consensus partition matrices (Bi-Cop am) for gene discovery.
2. Ailon, N., Charikar, M. and Newman, A. (2008). Aggregating inconsistent information: ranking and clustering. Journal of the ACM (JACM),

3. Ayad, H.G. and Kamel, M.S. (2008). Cumulative voting consensus method for partitions with a variable number of clusters. IEEE Transactions on Pattern Analysis and Machine Intelligence,
4. Bertolacci, M. and Wirth, A. (2007). Are Approximation Algorithms for Consensus Clustering Worthwhile? Proceedings of the Seventh SIAM International Conference on Data Mining, April 2007, SIAM, Minneapolis, MN.
5. Brannon, A.R., Reddy, A., Seiler, M. et al. (2010). Molecular stratification of clear cell renal cell carcinoma by consensus clustering reveals distinct subtypes and survival patterns.
6. Genes and Cancer. Cover, T.M. and Thomas, J.A. (2006). Elements of Information Theory, 2nd edition, Wiley-Interscience,
7. Dimitriadou, E., Weingessel, A. and Hornik, K. (2001). Voting-merging: an ensemble method for clustering, in Artificial Neural Networks–ICANN 2001, Springer,
8. Dudoit, S. and Fridlyand, J. (2003). Bagging to improve the accuracy of a clustering procedure. Bioinformatics,
9. Filkov, V. and Skiena, S. (2004). Integrating microarray data by consensus clustering. International Journal on Artificial Intelligence Tools,
10. Fred, A.L. and Jain, A.K. (2005). Combining multiple clustering using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence,
11. Ghaemi, R., Suleiman, N., Ibrahim, H. and Mustapha, N. (2009). A survey: clustering ensembles techniques. Proceedings of World Academy of Science: Engineering and Technology,
12. Gionis, A., Mannila, H. and Tsaparas, P. (2007). Clustering aggregation. ACM Transactions on Knowledge Discovery from Data (TKDD),
13. Li, T., Ogihara, M. and Ma, S. (2010). On combining multiple clustering: an overview and a new perspective. Applied Intelligence.
14. Mirkin, B. (1998). Mathematical Classification and Clustering: From How to What and Why, Springer, New York.
15. Monti, S., Tamayo, P., Mesirov, J. and Golub, T. (2003). Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. Machine Learning,
16. Rand, W.M. (1971). Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association,
17. Seiler, M., Huang, C.C., Szalma, S. and Bhanot, G. (2010a). Consensus Clustering Software Package, HTTP:// code.google.com/p/consensus-cluster/ (accessed 24 July 2014).
18. Seiler, M., Huang, C.C., Szalma, S. and Bhanot, G. (2010b). Consensus Cluster: a software tool forunsupervised cluster discovery in numerical data. OMICS A Journal of Integrative Biology,
19. Strehl, A. (2011). Cluster Analysis and Cluster Ensemble Software, http://strehl.com/download/cluster-PackV20.zip (accessed November 2014).
20. Strehl, A. and Ghosh, J. (2003). Cluster ensembles - a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research,
21. Topchy, A., Jain, A.K. and Punch, W. (2003). Combining Multiple Weak Clustering. ICDM 2003. Third IEEE International Conference on Data Mining, November 2003, Melbourne, FL, IEEE,
22. Topchy, A., Jain, A.K. and Punch, W. (2005). Clustering ensembles: models of consensus and weak partitions.IEEE Transactions on Pattern Analysis and Machine Intelligence,
23. Vega-Pons, S. and Ruiz-Shulcloper, J. (2011). A survey of clustering ensemble algorithms. International Journal of Pattern Recognition and Artificial Intelligence,
24. Yu, Z., Wong, H.-S. And Wang, H. (2007). Graph-based consensus clustering for class discovery from gene expression data. Bioinformatics,