Least squares consensus clustering: criteria, methods, experiments

Boris G. Mirkin and Andrey Shestakov

National Research University Higher School of Economics
bmirkin@hse.ru
shestakoffandrey@gmail.com
http://www.hse.ru

Abstract. We consider the problem of combining multiple clusterings of a set objects into one consolidated partition that is also referred to as consensus partition. In this work two algorithms based on the least squares criterion are introduced. To demonstrate the effectiveness our methods the results of two experimental sessions are provided. The first session shows that presented algorithms generate better consensus partitions, comparing with those of several other consensus clustering methods. The second session demonstrates that after multiple runs of k-means algorithm on one dataset consensus partitions outperform best partition according to k-means criterion.

Keywords: consensus, ensemble, aggregation, multiple partitions, cluster analysis, unsupervised learning, least squares

1 Introduction

Different clustering algorithms or different runs of the same algorithm can provide different partitions of the same dataset. For example, this can be clearly seen after applying k-means algorithm with different initialized centroids. The subject of this paper is the problem of aggregation multiple clusterings into a single consolidated clustering – consensus partition. This process is known in the literature as clustering ensembles, clustering aggregation or consensus clustering [1]. There are many reasons for using such techniques. First of all, consensus clustering is expected to provide more accurate and stable solutions on average, compared to the conventional single clustering approaches. Single clustering algorithms may be ineffective for certain datasets. Usually, by combining different clusterings together one may obtain a better partition by taking into account the biases of individual solutions.

Perhaps the grand start for consensus clustering approach was made by A. Strehl and J. Ghosh in paper [5]. Since then Consensus clustering has become quite popular in recent years. Several applications can be found in bioinformatics, particulary in gene expression analysis [12–17], web-document clustering [19] and categorical data analysis [20]. According to [1] consensus clustering algorithms can be organized in three main categories: probabilistic approach, direct approaches and pairwise similarity-based approach. Mixture models [6] and Bayesian consensus clustering model [7] form the base of the first approach. Consensus mechanisms of the direct approach operates with graph models [5] and voting procedures [8, 11, 10]. Algorithms in pairwise similarity-based approach take modifications of coassociation matrix as input parameter [9, **?**,4]. Entity a_ij of coassociation matrix $A = (a_{ij})$ shows the number of partitions in which two objects y_i and y_j are joined.

In his work [2] Mirkin proposed two symmetric least squared criteria to produce consensus partitions — ensemble and combined consensus. In this paper we set experiments, firstly, to compare partitions retrieved by least squared criteria and results of other well-known consensus methods, and secondly, to find the better way to process multiple runs of k-means algorithm — with consensus or with k-means criterion. The paper structured as follows. !!!!!!!To Do!!!!!!!!

2 Least squared criteria

Let $Y = \{y_1, \ldots, y_N\}$ be a set on N objects. Partitioning of dataset Y on K non-overlapping classes $S = \{S_1, \ldots, S_K\}$ can be represented by binary membership $N \times K$ matrix $Z = (z_{ik})$ such that $z_{ik} = 1$ if y_i belongs to S_k and $z_{ik} =$ otherwise (see example on fig.1).

$$S = \{S_1, S_2, S_3\} = \begin{matrix} y_1 : \\ y_2 : \\ y_3 : \\ y_4 : \\ y_5 : \\ y_6 : \end{matrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \\ 2 \\ 2 \end{bmatrix} \Leftrightarrow Z =$$

	S_1	S_2	S_3
y_1 :	1	0	0
y_2 :	0	1	0
y_3 :	0	0	1
y_4 :	1	0	0
y_5 :	0	1	0
y_6 :	0	1	0

Fig. 1. Example of membership matrix

Consider matrices $Z^T Z$ and ZZ^T with sizes $K \times K$ and $N \times N$ respectively. Diagonal elements of the first matrix are equal to classes cardinalities N_k, whereas elements of the secound are equal to 1 if two objects are joined in class and 0 otherwise. Hence, matrix $P_Z = Z(Z^T Z)^{-1} Z^T = (p_{ij})$ is $N \times N$ matrix, with $p_{ij} = \frac{1}{N_k}$, if $\{y_i, y_j\} \in S_k$ and 0 otherwise. Matrix P_Z is also known as projection matrix on linear subspace $L(Z)$. Projection matrices posses idempotence and symmetric properties.

2.1 Ensemble consensus

The main idea of ensemble consensus is to find such partition S that can best predict clusters in given profile of T partitions $R = \{R^1, R^2, \ldots, R^T\}$. Consider membership matrices X^t, $t = 1, \ldots, T$ for partitions in given profile R. Similarly binary matrix Z represents consensus partition S. Hence, each of given partitions is related to target partition S as follows:

$$x_{il}^t = \sum_{k=1}^{K} c_{kl}^t z_{ik} + e_{ik}^t, \tag{1}$$

where coefficients c_{kl}^t and matrix elements z_{ik} are chosen to minimize residuals e_{ik}^t. In matrix notations criterion to minimize can be reformulated as

$$E^2 = \|X - P_Z X\|^2, \tag{2}$$

where X is concatenation of matrices $X^1, \ldots, X_t$ and $\| \cdot \|^2$ denotes the sum of squares of the matrix elements. Expanding (2) brings us to the following criterion:

$$g(S) = \sum_{k=1}^{K} \sum_{i,j \in S_k} \frac{a_{ij}}{N_k}, \tag{3}$$

where a_{ij} is element of consensus matrix A which is used in pairwise similarity-based approach.

2.2 Combined consensus

The aim of combined consensus is symmetrical to emsemble consensus: find such partition S which is best predicted by given profile of partitions $R = \{R^1, R^2, \ldots, R^T\}$. Hidden partition Z is related to given partitions $R^1, R^2, \ldots, R^T$ as

$$z_{ik} = \sum_{l=1}^{L} c_{kl}^u x_{il} + e_{ik}^u, \tag{4}$$

where coefficients c_{kl}^u and matrix elements z_{ik} are chosen to minimize residuals e_{ik}^u. Similarly to section 2.1, matrix formulation of criterion to minimize is

$$E^2 = \sum_{u=1}^{T} \|Z - P_u Z\|^2. \tag{5}$$

which in turn can be written as

$$f(S, \overline{p}) = \sum_{k=1}^{K} \sum_{i,j \in S_k} (p_{ij} - \overline{p}), \tag{6}$$

where $\overline{p} = T/N$.

3 Algorithms

3.1 AddRemAdd(j) algorithm

Algorithm AddRemAdd(j) was proposed by Mirkin in [2] and represents the approach or one-by-one clustering. Applied to each object y_j that method outputs cluster with maximum within cluster similarity according to matrix A. AddRemAdd(j) runs in a loop over all $j = 1 \dots N$. When it results in cluster $S(j)$ AddRemAdd is applied on $Y' = Y/S(j)$ with reduced matrix A' which is related to elements in Y' and stops when no unclustered entities remain.

Input: Consensus coassosiation matrix A
Output: Cluster $S(j)$ which includes y_j and correspondig λ (the average similarity of each entity to $S(j)$) and g^2 (the contribution)

1. $n := 1$, $N-$dimensional $z := -1$ exept $z_j := 1$
   For each x_i compute $a(i, S) = a_{ij}$
2. Find i^*, which maximizes $a(i, S)$)
   if $a(i^*, S) > \lambda/2$
   - $z_{i^*} = -z_{i^*}$
   - Update
     $\lambda = \frac{(n-2)\lambda + 2z_{i^*}a(i^*,S)}{n}$
     $a(i, S) = \frac{(n-1)a(i^{\cdot}S) + z_i * a_{ii^*}}{n}$
     $g^2 = \lambda^2 n^2$
   - repeat **step 2**

   else
   - Output $S(j)$, λ g^2

Emsemble consensus partition consists of the AddRemAdd cluster outputs: $S^* = \bigcup S(j)$

3.2 Agglomerative Algorithm

Starting from singleton partition $S = \{S_1 = \{y_1\}, \dots, S_N = \{y_N\}\}$, each iteration of algorithm joins two clusters, such that obtained partition maximizes criterion (6).

Input: Projection matrix $\overline{P}$
Output: Combined Consensus S^*

1. Starting from singleton partition $S(N)$
2. Find $p_{rt} = max(\overline{P})$, $\quad r \neq t$
   if $p_{rt} > 0$
   - Join S_r and S_t
   - Update $\overline{P'}$ by componentwise summation of rows and columns with indexes r and t
   - Repeat **step 2** for $\overline{P'}$

   else
   - Assign last obtained partition as Combined Consensus Partition S^*

4 Experiments

All evaluations are done on synthetic datasets that have been generated using Netlab library [18]. Dataset consists of 1000 twelve-dimensional objects comprising nine spherical Gaussian clusters. The variance of each cluster lies in $0.1 - 0.3$ and its center's are independently generated from Gaussian distribution $\mathcal{N}(0, 0.7)$.
Let us denote generated partition as Λ with $k_\Lambda = 9$ clusters. The profile of partitions $R = \{R^1, R^2, \ldots, R^T\}$ for consensus algorithms is constructed as result of $T = 50$ k-means runs over generated dataset. Depending on k in k-means we carry out experiment in four settings: a) $k = 9 = k_\Lambda$, b) $k = 6 < k_\Lambda$, c) $k = 12 > k_\Lambda$, d) k is uniformly random on $[6, 12]$. After applying consensus algorithms, Adjusted Rand Index (ARI) [1] for consensus partitions S and generated partition Λ is computed. $\varphi^{ARI}(S, \Lambda)$ is a measure of the similarity between two partitions: it equals 1 if partitions are identical.

4.1 Comparing consensus algorithms

Least squares consensus have been compared with the following algorithms:

- Voting Scheme (Dimitriadou, Weingessel and Hornik - 2002) [8]
- cVote (Ayad - 2010) [11]
- Fusion Transfer (Guenoche - 2011) [9]
- Borda Consensus (Sevillano, Carrie and Pujol - 2008) [10]
- Meta-CLustering Algorithm (Strehl and Ghosh - 2002) [5]

Each setting of experiment a)-d) was launched 10 times. Averaged results are presented in respective Tables 1-4

Table 1. $\phi^{ARI}(S, \Lambda)$ and number of classes if $k_\Lambda = k = 9$

Algorithm	Average ϕ^{ARI}	Std. ϕ^{ARI}	Avr. # of classes	Std. # of classes
Aggl	**0.9106**	0.0398	6	0.6667
ARA	**0.9578**	0.0246	7.6	0.5164
Vote	0.7671	0.0624	8.9	0.3162
cVote	0.7219	0.0882	8.1	0.7379
Fus	0.7023	0.0892	11.6	1.8379
Borda	0.7938	0.1133	8.5	0.7071
MCLA	0.7180	0.0786	8.6	0.6992

Table 2. $\phi^{ARI}(S, \Lambda)$ and number of classes if $k_\Lambda > k = 6$

Algorithm	Average ϕ^{ARI}	Std. ϕ^{ARI}	Avr. # of classes	Std.# of classes
Aggl	0.7374	0.1083	4.8	0.4216
ARA	**0.8333**	0.0586	6.2	0.6325
Vote	0.7769	0.0895	5.9	0.3162
cVote	0.7606	0.0774	5.6	0.6992
Fus	**0.8501**	0.1154	7.7	1.3375
Borda	0.7786	0.0916	6	0
MCLA	0.7902	0.0516	6	0

Table 3. $\phi^{ARI}(S, \Lambda)$ and number of classes if $k_\Lambda < k = 12$

Algorithm	Average ϕ^{ARI}	Std. ϕ^{ARI}	Avr. # of classes	Std.# of classes
Aggl	**0.9353**	0.0458	6.6	0.5164
ARA	**0.9729**	0.0313	9	0.9428
Vote	0.6958	0.0796	11.4	0.5164
cVote	0.672	0.0887	10.9	0.7379
Fus	0.6339	0.0827	16	4
Borda	0.7132	0.074	11.1	0.7379
MCLA	0.6396	0.0762	11.9	0.3162

Table 4. $\phi^{ARI}(S, \Lambda)$ and number of classes if $k \in [6, 12]$

Algorithm	Average ϕ^{ARI}	Std. ϕ^{ARI}	Avr. # of classes	Std.# of classes
Aggl	**0.9230**	0.0323	5.4	1.075
ARA	**0.9648**	0.019	6.8	0.7888
cVote	0.5771	0.1695	10.4	1.2649
Fus	0.62	0.0922	11.6	2.0656
MCLA	0.6567	0.1661	10.6	1.3499

Tables above consistently show the following:

- Algorithms based on least squares criteria have outperformed other consensus algorithms — average ϕ^{ARI} is higher while it's standard deviation is closer to zero
- The only exception is stage c) with $k_\Lambda > k = 6$ (see table 2 where Fusion Transfer algorithm demonstrated a little better results probably because of transfer procedure
- It can be noticed, that average number of classes in produced consensus clusterings is lower than k in profile R and k_Λ

4.2 Comparing consensus and k-means criterion

Target function of k-means algorithm is the following:

$$J(R) = \sum_{i=1}^{k} \sum_{j \in R'_i} d(c_i, y_j), \tag{7}$$

where c_i is i's cluster centroid of partition $R' = \{R'_1, \ldots, R'_k\}$, and $d(\cdot)$ is similarity measure. In this experiment results of Agglomerative and AddRemAdd algorithms are compared with best partition of k-means in profile R according to criteria (7). We also built profile R according to settings a), that is $k = k_\Lambda$ and d), that is $k \in [6, 12]$. The remaining scheme of experiment is the same as in the previous one. Averaged results are presented in tables 5-6

Table 5. $\phi^{ARI}(S, \Lambda)$ and number of classes if $k = 9 = k_\Lambda$

Algorithm	Average ϕ^{ARI}	Std. ϕ^{ARI}	Avr. # of classes	Std.# of classes
Best k-means	0.7548	0.0829	9	0
Aggl	**0.8448**	0.0605	6	0.6667
ARA	**0.8929**	0.0459	7.9	1.1005

Table 6. $\phi^{ARI}(S, \Lambda)$ and number of classes if $k \in [6, 12]$

Algorithm	Average ϕ^{ARI}	Std. ϕ^{ARI}	Avr. # of classes	Std. # of classes
Best k-means	0.5955	0.0591	12	0
Aggl	**0.8180**	0.1153	5.8	0.6325
ARA	**0.8684**	0.0763	7.9	1.1005

Based on tables above one can conclude on average consensus outperforms best k-means partitions, however the issue of lower number of classes is still remains.

5 Conclusions

Consensus clustering approach is becoming even more popular. In this paper we demonstrated that criteria besed on least squares, that have been formulated by Mirkin in 1980, are competitiveness to other modern consensus methods: they provide partitions, which are closer to the generated partitions than the consensus partitions found with the other techniques. The second contribution of this paper is experimentally demonstrated effectiveness of consensus over k-means criterion in task of processing results of multiple k-means runs.

References

1. J. Ghosh, A. Acharya. "Cluster ensembles". - Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2011.
2. B. Mirkin. Analysis of Qualitative Attributes and Structures, - Moscow, 1980.
3. B. Mirkin. Core concepts in Data Analysis: summarization, correlation, visualization. - Springer, 2011.
4. B. Mirkin. Clustering: A Data Recovery Approach, 2012 (to appear).
5. A. Strehl, J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. - Journal on Machine Learning Research, 2002.
6. A. Topchy, A. K. Jain, and W. Punch A mixture model for clustering ensembles. In Proceedings SIAM International Conference on Data Mining, 2004.
7. H. Wang, H. Shan, A. Banerjee Bayesian cluster ensembles. In: Proceedings of the Ninth SIAM International Conference on Data Mining, 211222, 2009.
8. E. Dimitriadou, A. Weingessel and K. Hornik. A Combination Scheme for Fuzzy Clustering. - Journal of Pattern Recognition and Artificial Intelligence, 2002.
9. A. Guenoche. Consensus of partitions : a constructive approach. - Adv. Data Analysis and Classification 5, pp. 215-229, 2011.
10. X. Sevillano Dominguez, J. C. Socoro Carrie and F. Alias Pujol. Fuzzy clusterers combination by positional voting for robust document clustering. - Procesamiento del lenguaje natural, 43, pp. 245-253.
11. H. Ayad, M. Kamel On voting-based consensus of cluster ensembles. - Pattern Recognition, pp. 1943-1953, 2010.
12. S. Swift A. Tucker, V. Vinciotti, N. Martin, C. Orengo, X. Liu, P. Kellam. Consensus clustering and functional interpretation of gene-expression data. - Genome Biology, 2004.
13. S. Monti, P. Tamayo, J. Mesirov, T. Golub. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. - Machine Learning, 2003.
14. A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay. Multiobjective Genetic Clustering with Ensemble Among Pareto Front Solutions: Application to MRI Brain Image Segmentation - Seventh International Conference on Advances in Pattern Recognition, 2009.
15. Chia-Wei Chu, J. D. Holliday, P. Willett. Combining multiple classifications of chemical structures using consensus clustering. - Bioorganic and Medicinal Chemistry, 2012.
16. Chia-Wei Chu, J. D. Holliday, P. Willett. Analyzing High Dimensional Toxicogenomic Data Using Consensus Clustering. - Bioorganic and Medicinal Chemistry, 2012.
17. C. Gao, D. Weisman, N. Gou, V. Ilyin and A. Z. Gu. Analyzing High Dimensional Toxicogenomic Data Using Consensus Clustering. - Environmental Science and Technology, 2012.
18. Netlab Neural Network software, ¡http://www.ncrg.aston.ac.uk/netlab/index.php¿
19. Sevillano X, Cobo G, Al'as F, Socor 'o JC. Feature diversity in cluster ensembles for robust document clustering. - Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006.
20. Zengyou He, Xiaofei Xu, Shengchun Deng. A cluster ensemble method for clustering categorical data. - Information Fusion, 2005.