Multi-view Clustering Ensembles

Xijiong Xie, Shiliang Sun

Department of Computer Science and Technology, East China Normal University
500 Dongchuan Road, Shanghai 200241, P. R. China
E-MAIL: xjxie11@gmail.com, slsun@cs.ecnu.edu.cn

Abstract:

Multi-view clustering and clustering ensembles have become increasingly popular in recent years. Multi-view clustering employs relationship of views to cluster data and clustering ensembles combine different component clusterings to a better final partition. In this paper, we proposed multi-view clustering ensembles which extend clustering ensembles to multi-view clustering. Experimental results show the good performance of multi-view spectral clustering ensembles and multi-view kernel k-means clustering ensembles on real datasets.

Keywords:

multi-view clustering; clustering ensembles; spectral clustering; kernel k-means clustering

1. Introduction

Clustering is a key issue in intelligence science and is widely used in the field of artificial intelligence. The technique has been studied for several decades in areas of pattern recognition, machine learning, applied statistics, communications and information theory. It is applied to numerous fields of applications including data mining, text mining, bio-informatics, image analysis and segmentation, data compression, and data classification.

Multi-modal datasets are very common in practice because of the use of different measuring methods (e.g, infrared and visual cameras), or of different media, like text, video and audio. Each instance has multiple representations, called views. The natural and frequent occurrence of multi-view data has raised interest in the so called multi-view learning [1]. Multi-view clustering explores and exploits multiple views simultaneously in order to obtain a more accurate and robust partitioning of the data than single view clustering. There exist two methods in multi-view clustering: centralized and distributed [6]. Centralized algorithms simultaneously use all views to cluster the data while distributed algorithms cluster each view independently from others, using a single view algorithm, and then combine the individual clustering to obtain a final partitioning. During the past decade, Bickel and Scheffer [2] developed a two-view EM and a two-view spherical k-means algorithm under the assumption that the views are independent. De Sa [3] proposed a two-view spectral clustering algorithm that creates a bipartite graph and is based on the "minimizing-disagreement" idea. Kumar et al. [4] proposed a co-training approach for multi-view spectral clustering and co-regularized multi-view spectral clustering [5]. Grigorios Tzprtzis proposed convex mixture models for multi-view clustering [6] and kernel-based weighted multi-view clustering [7].

Clustering ensembles learning is an active research hotspot and is regarded as an important research branch in machine learning field. The detailed representation of clustering ensembles was firstly proposed by Strehl and Ghosh [8]. The nature of clustering ensembles is that different component clusterings are combined to a better final partition via a consensus function. A diversity of component clusterings can be obtained by a number of approaches, such as using different conventional algorithms, their relaxed versions and built-in randomness, or by data sampling. Selective ensemble have been attracted many researchers in the supervised area while few papers have considered the unsupervised ensemble. Zhou and Tang proposed an ensemble approach based on bagging [9, 10]. Fern and Lin [11] designed three ensemble selection methods based on quality and diversity. Hong et al. [12] introduced a novel selective clustering ensemble method through resampling. Recently, Azimi and Fern [13] proposed an adaptive cluster ensembles method.

However, designing a well consensus function is important to clustering ensembles. Fred [14] proposed to integrate various component clusterings through a co-association matrix, which represents the frequency of each pair of samples appearing in the same cluster. The final result of clustering en-

sembles is obtained using a voting-type method applied to the co-association matrix. Strehl and Ghosh [8] have developed three different consensus functions based on hypergraph for ensemble learning: cluster-based similarity partitioning algorithm, hypergraph-partitioning algorithm, and meta clustering algorithm. All of them addressed various hypergraph operation to find a solution. Topchy et al. [15] designed a consensus function based on a finite mixture model. The final partition is found as a solution to a maximum likelihood problem for a given clustering ensembles. In this paper, we address clustering ensembles based on selective voting which work by measuring the similarity between the clusters through counting their overlapped data items [9].

The rest of this paper is organized as follows. Section 2 introduces kernel k-means clustering, spectral clustering, multi-view spectral clustering and multi-view kernel k-means clustering. Section 3 introduces method of clustering ensembles we employ and multi-view clustering ensembles. After reporting experimental results in Section 4, we give conclusions and future work in Section 5.

2. Multi-view kernel k-means clustering and multi-view spectral clustering

2.1. Kernel k-means

Kernel k-means is a generalization of the standard k-means algorithm where the dataset $\chi = \{x_i\}_{i=1}^N, x_i \in R^d$ is mapped to a higher dimensional reproducing kernel Hilbert space via the use of kernel trick.

In order to partition dataset χ into M disjoint clusters, $\{C_k\}_{k=1}^M$, the intra-cluster variance in feature space is represented by

$$\varepsilon_H = \sum_{i=1}^{N}\sum_{k=1}^{M}\delta_{jk}\|\phi(x_i) - m_k\|^2, m_k = \frac{\sum_{i=1}^N \delta_{jk}\phi(x_i)}{\sum_{i=1}^N \delta_{jk}}, \quad (1)$$

which is minimized over clusters $\{C_k\}_{k=1}^M$, where m_k is the k-th cluster center and δ_{ik} is an indicator variable with $\delta_{ik} = 1$ if $x_i \in C_k$ and 0 otherwise. Through defining transformation ϕ, kernel function $K \in R^{N\times N}$ can be written as $K_{ij} = K(x_i, x_j) = \phi(x_i)^T\phi(x_j)$ which is the most common way of representing data in feature space. The squared Euclidean distances in (1) can be computed using solely the kernel matrix entries which are written as

$$\|\phi(x_i) - m_k\|^2 = K_{ii} - \frac{2\sum_{i=1}^N \delta_{jk}K_{ij}}{\sum_{i=1}^N \delta_{jk}} + \frac{\sum_{j=1}^N\sum_{l=1}^N \delta_{jk}\delta_{lk}K_{jl}}{\sum_{j=1}^N\sum_{l=1}^N \delta_{jk}\delta_{lk}} \quad (2)$$

(centers m_k cannot be analytically calculated).

Kernel k-means monotonically converges to a local minimum by iteratively updating the partitioning through assignments of the instances to their closest center in feature space, which heavily depends on the initial cluster assignments. The deterministic-incremental approaches such as the global kernel k-means algorithm could be applied in order to overcome this disadvantage.

2.2. Spectral clustering

The intra-cluster variance of spectral clustering [16] can be equivalently written as a trace difference:

$$\varepsilon_H = tr(K) - tr(Y^T KY), \quad (3)$$

where $Y \in R^{N\times M}, Y_{ik} = \frac{\delta_{ik}}{\sqrt{\sum_{j=1}^N \delta_{jk}}}$. The first term on the above equation is a constant, so the minimization of (3) is equivalent to the maximization of $tr(Y^T KY)$. If Y is relaxed to be an arbitrary orthonormal matrix. The optimal Y is consists of the top M eigenvectors of the kernel matrix K.

2.3. Multi-view extensions

In this section, we briefly introduce multi-view spectral clustering (MvSpec) and multi-view kernel k-means clustering (MvKKM) [7]. They constitute the foundation of our subsequent proposed methods. The methods addressed kernel-based scheme which embeds in the clustering process an automatic "ranking" of the views. On kernel learning, this exploits kernels as a tool for representing and combining views in multi-view learning. We suppose a dataset χ with N instances and V views:$\chi = \{x_i\}_{i=1}^N$, where $x_i = \{x_i^{(v)}\}_{v=1}^V$. Through kernel methods, the dataset is implicitly mapped to a feature space and is represented by V kernel matrix $\{K^{(v)}\}_{v=1}^V$, and composite

kernels can be represented by

$$\widetilde{K} = \sum_{v=1}^{V} w_v^p K^{(v)}, w_v \geq 0, \sum_{v=1}^{V} w_v = 1, p \geq 1. \tag{4}$$

The objection function of multi-view kernel k-means can be written as

$$\begin{aligned} &\varepsilon_{\widetilde{H}} = \sum_{v=1}^{V} w_v^p \sum_{i=1}^{N} \sum_{k=1}^{M} \\ &\delta_{ik} \|\phi^{(v)}(x_i^{(v)}) - m_k^{(v)}\|^2, \\ &m_k^{(v)} = \frac{\sum_{i=1}^{N} \delta_{ik} \phi^{(v)}(x_i^{(v)})}{\sum_{i=1}^{N} \delta_{jk}}. \end{aligned} \tag{5}$$

The optimization problem for multi-view kernel k-means can be written as

$$\begin{aligned} &\min_{\{w_v\}_{v=1}^{V}} \varepsilon_{\widetilde{H}} \\ &\text{s.t. } w_v \geq 0, \sum_{v=1}^{V} w_v = 1, p \geq 1. \end{aligned} \tag{6}$$

Under the spectral perspective, the objection function of multi-view spectral clustering can be written as in terms of matrix traces

$$\begin{aligned} \varepsilon_{\widetilde{H}} &= tr(\widetilde{K}) - tr(Y^T \widetilde{K} Y) \\ &= \sum_{v=1}^{V} w_v^p (tr(K^{(v)}) - tr(Y^T K^{(v)} Y)), \end{aligned} \tag{7}$$

where $K^{(v)}$ is a positive semidefinite matrix and $Y^T Y = I, Y \in R^{N \times M}$. The optimization problem for multi-view spectral clustering can be written as

$$\begin{aligned} &\min_{\{w_v\}_{v=1}^{V}} \varepsilon_{\widetilde{H}} \\ &\text{s.t. } w_v \geq 0, \sum_{v=1}^{V} w_v = 1. \end{aligned} \tag{8}$$

3. Multi-view clustering ensembles

3.1. Clustering ensembles

Suppose that $X = \{x_1, x_2, \cdots, x_n\} \subset R^d$ denotes an unlabeled dataset. The set is partitioned H times by clustering algorithms to get H component clustering results $\Pi =$

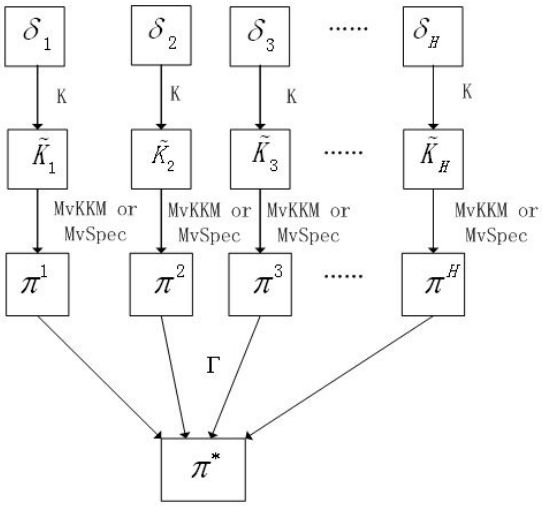

Figure 1. A general framework for multi-view clustering ensembles

$\{\pi^1, \pi^2, \cdots, \pi^H\}$, where $\pi^i (i = 1, 2, \cdots, H)$ is the clustering algorithm results of the ith run. In clustering ensemble, H component clusterings are combined to final clustering result π^* by a consensus function τ [17]. The clusterings are aligned based on the recognition that similar clusters should contain similar data items [9]. For example, suppose there are two clusterings whose corresponding label vectors π^a and π^b and each clustering divide the dataset into k clusters, such as $\{C_1^a, C_2^a, \cdots, C_k^a\}$ and $\{C_1^b, C_2^b, \cdots, C_k^b\}$. For C_i^a and C_j^b, the number of overlapped data items which appear both clusters is counted. Then, the pair of clusters whose number of overlapped data items is the largest, are matched in the way that they are denoted by the same label. Such a process is repeated until all the clusters are matched. There are many methods combining the H component clustering results. The method we use is selective voting, where the ith component of the label vector corresponding to the ensemble. For example, π_i is determined by the plurality voting result of $\{\pi_i^1, \pi_i^2, \cdots, \pi_i^H\}$.

3.2. Multi-view extensions

In this section, we combine clustering ensembles with multi-view clustering. As mentioned before, the methods of multi-view kernel k-means clustering and multi-view spectral clustering are based on kernel trick. Radial basis function kernel

(RBF) can be written as

$$K(x_i, x_j) = exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2}), \quad (9)$$

where σ represents the scale parameter. Because the scale parameter in RBF kernel can be different, we can design that composite kernels $\widetilde{K}$ are different. In multi-view kernel k-means clustering and multi-view spectral clustering, different σ can get different component clustering results. Addressing the advantage, we can combine clustering ensembles with multi-view clustering. Then we can see a general framework for multi-view clustering ensembles in the Fig.1. We described multi-view kernel k-means clustering ensembles (MvKKMCE) and multi-view spectral clustering ensembles (MvSpecCE) in Algorithm 1 and Algorithm 2 respectively. Here 'ClusterEnsemble' represents clustering ensembles algorithm and 'n' represents the number of clustering ensembles. 'v' represents the number of views. 'label' represents the true label of clusters. 'clusters' represents the number of clusters.

Algorithm 1 Multi-view kernel k-means clustering ensembles

Input: v-view data X, label, clusters, n
for each $\sigma_i \, in[\sigma_{min}, \sigma_{max}]$ and $0 \leq i \leq n$ **do**
  for $j = 1$ **to** v **do**
    $K^{(v)}$
  end for
  $\tilde{K_i} = [K_i^{(1)}; K_i^{(2)}; \cdots K_i^{(v)}]$
  $[\text{Clusterelem}_i] = \text{MvKKM}(\tilde{K_i}, \text{clusters})$
end for
$[\text{Clusterelem}] = [\text{Clusterelem}_1; \text{Clusterelem}_2;$
$\cdots ; \text{Clusterelem}_n]$
$\text{mico-p} = \text{ClusterEnsemble}(\text{Clusterelem}, \text{label}, n)$

Algorithm 2 Multi-view spectral clustering ensembles

Input: v-view data X, label, clusters, n
for each $\sigma_i \, in[\sigma_{min}, \sigma_{max}]$ and $0 \leq i \leq n$ **do**
  for $j = 1$ **to** v **do**
    $K^{(v)}$
  end for
  $\tilde{K_i} = [K_i^{(1)}; K_i^{(2)}; \cdots ; K_i^{(v)}]$
  $[\text{Clusterelem}_i] = \text{MvSpec}(\tilde{K_i}, \text{clusters})]$
end for
$[\text{Clusterelem}] = [\text{Clusterelem}_1; \text{Clusterelem}_2;$
$\cdots ; \text{Clusterelem}_n]$
$\text{mico-p} = \text{ClusterEnsemble}(\text{Clusterelem}, \text{label}, n)$

Table 1. Datasets.

Name	Attributes	Instances	Classes
Ionosphere	34	351	2
Handwritten digits data	649	2000	10

In this paper, we address micro-precision as the evaluation of clustering performance. The clusterings are transformed into classifiers using the following method: identify each cluster with the class that has the largest overlap with the cluster, and assign every data item in that clustering to the found class. The method requires multiple clusters to be assigned to a single class, but never assigns a single cluster to multiple classes. We suppose there are c classes, i.e. $\{C_1, C_2, \cdots, C_c\}$, in the ground truth classification. For a given clustering, by using the above method, let a_t denote the number of data items that are correctly assigned to the class C_t. 'm' represents the number of examples. Then, the clustering performance can be measured by micro-precision as

$$\text{mico-p} = \frac{1}{m} \sum_{t=1}^{c} a_t. \quad (10)$$

The bigger the value of micro-p, the better the clustering performance.

4. Experimental results

In this section, we design two experiments based on Ionosphere dataset and handwritten digits dataset which come from UCI Machine Learning Repository. We compare the performance of MvKKMCE with the performance of MvSpecCE on the two datasets. Details about the two datasets are listed in Table 1.

4.1. Ionosphere

The ionosphere dataset was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. The targets were free electrons in the ionosphere."Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not and their signals pass through the ionosphere. It includes 351 instances in total which are divided into 225 "Good" (positive) instances and 126 "Bad" (negative) instances.

Table 2. Clustering result (n=10) (%) on Ionosphere dataset.

Method	p=1	p=1.3	p=1.5	p=2	p=4	p=6
MvSpecCE	80.63	81.77	80.91	73.22	77.21	75.50
MvKKMCE	**86.32**	**86.04**	**86.04**	**86.04**	**86.04**	**86.04**

Table 3. Clustering result (n=5) (%) on handwritten digits(0$\sim$ 4).

Method	p=1	p=1.3	p=1.5	p=2	p=4	p=6
MvSpecCE	77.40	82.60	86.30	95.80	96.60	96.80
MvKKMCE	**82.40**	**89.50**	**94.40**	**96.60**	**97.20**	**97.30**

In our experiments, we regard original data as the first view. Then we capture 99% of the data variance while reducing the dimensionality from 34 to 21 with PCA and regard dealt data as the second view. We take a coarse grid search for scale parameter σ in the region $[2^0, 2^6]$ with with exponent growth 0.5. Then we take the finer grid search on the neighborhood of the best results of the coarse search for ten times. So it can avoid doing more consuming exhaustive parameter search. Because p parameters have impact on MvKKM and MvSpec, the proposed algorithms are executed for various p values. From Table 2, we can conclude that MvKKMCE are much better than MvSpecCE as p increases.

4.2. Handwritten digits

This dataset consists of features of handwritten digits $(0 \sim 9)$ extracted from a collection of Dutch utility maps. It consists of 2000 examples (200 examples per class) with view-1 being the 76 Fourier coefficients, and view-2 being the 216 profile correlations of each example image.

In our experiments, we do two experiments on the same data sets and evaluate our model with $(0 \sim 4)$ and $(5 \sim 9)$ respectively five clusters. The attributes are normalized to unit variance as attributes within the same view exhibit significantly different scales. We take a coarse grid search for scale parameter σ in the region $[2^0, 2^6]$ with exponent growth 0.5. Then we take the finer grid search on the neighborhood of the best results of the coarse search for five times. From Table 3 and Table 4, we can conclude that MvKKMCE are almost always better than MvSpecCE as p increases.

Table 4. Clustering result (n=5) (%) on handwritten digits(5$\sim$ 9).

Method	p=1	p=1.3	p=1.5	p=2	p=4	p=6
MvSpecCE	77.40	82.60	86.30	94.00	96.80	96.90
MvKKMCE	71.40	**92.60**	**93.30**	93.50	**97.30**	**97.30**

5. Conclusions

In this paper, we proposed multi-view clustering ensembles based on multi-view clustering and clustering ensembles. We compare the performance of MvKKMCE with the performance of MvSpecCE on two real datasets and conclude that the performance of MvKKMCE is almost always better than the performance of MvSpecCE. In the future, it would be interesting to extend clustering ensembles to other multi-view clustering algorithms.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Project 61075005, and Shanghai Knowledge Service Platform Project (No. ZF1213).

References

[1] S. Sun,"A survey of multi-view machine learning", Neural Computing and Applications, pp. 1-8, 2013

[2] S. Bickel and T. Scheffer, "Multi-view clustering", Proceedings of the 4th IEEE International Conference on Data Mining, pp. 19-26, 2004

[3] V.R. De Sa,"Spectral clustering with two views", Proceedings of the 22th IEEE International Conference on Machine Learning, pp. 20-27, 2005

[4] A. Kumar and H. Daume, "A co-training approach for multiview spectral clustering", Proceedings of the 28th IEEE International Conference on Machine Learning, pp. 393-400, 2011

[5] A. Kumar, P. Rai and H. Daume, "Co-regularized Multi-view Spectral Clustering", Proceedings of the 12th IEEE International Conference on Data Mining, pp. 675-684, 2012

[6] G. Tzortzis and A. Likas, "Convex mixture models for multi-view clustering", Lecture Notes in Computer Science, pp. 205-214, 2009

[7] G. Tzortzis and A. Likas, "Kernel-based Weighted Multi-view Clustering", Proceedings of the 12th IEEE International Conference on Data Mining, pp. 675-684, 2012
[8] A. Strehl and J. Ghosh, "Cluster Ensembles-A Knowledge Reuse Framework for Combining Multiple Partitions", Journal of Machine Learning Research, pp. 583-617, 2002
[9] Z. Zhou and W. Tang, "Clusterer ensemble", Knowledge-Based Systems, pp. 77-83, 2006
[10] J. Jia, X. Xiao, B. Liu and L. Jiao, "Bagging-based spectral clustering ensemble selection", Pattern Recognition Letters, pp. 1456-1467, 2011
[11] X. Fern and L. Wei, "Cluster ensemble selection", Statistical Analysis and Data Mining, pp. 128-141, 2008
[12] Y. Hong, S. Kwong, H. Wang and Q. Ren, "Resampling-based selective clustering ensembles", Pattern Recognition Letters, pp. 298-305, 2009
[13] J. Azimi and X. Fern, "Adaptive cluster ensemble selection", Proceedings of the 21th IEEE International Joint Conference on Artificial Intelligence, pp. 992-997, 2009
[14] A. Fred, "Finding consistent clusters in data partitions", Proceedings of the 2rd IEEE International Workshop on Multiple Classifier Systems, pp. 309-318, 2001
[15] A. Topchy, A. Jain and W. Punch, "A mixture model for cluster ensembles", Proceedings of 4th IEEE SIAM International Conference on Data Mining, pp. 379-390, 2004
[16] A. Ng, M. Jordan and Y. Weiss, "On spectral clustering: Analysis and an algorithm", Advanced in Neural Information Processing Systems, pp. 849-856, 2001
[17] A. Topchy, K. Jain and W. Punch, "Cluster ensembles: Models of consensus and weak partitions.", IEEE Transaction on Pattern Analysis and Machine Intelligence, pp. 1866-1881, 2005