Multiview and multifeature spectral clustering using common eigenvectors

Samir Kanaan-Izquierdo[*, a, c]
samir.kanaan@upc.edu

Andrey Ziyatdinov[b]

Alexandre Perera-Lluna[a, c]

[a]Centre de Recerca en Enginyeria Biomèdica, Universitat Politècnica de Catalunya, Pau Gargallo 5, Barcelona, 08028, Spain

[b]Department of Epidemiology, Harvard T.H. Chan School of Public Health, Boston, Massachusetts, United States

[c]CIBER of Bioengineering, Biomaterials and Nanomedicine (CIBER-BBN), Barcelona, Catalonia, Spain

*Corresponding author at: Centre de Recerca en Enginyeria Biomèdica, Universitat Politècnica de Catalunya, Pau Gargallo 5, Barcelona, 08028, Spain. ~~Tel.: +34 934137389;fax: +34 93 413 74 00.~~

Abstract

An ever-increasing number of data analysis problems include more than one view of the data, i.e. different measurement approaches to the population under study. In consequence, pattern analysis methods that deal appropriately with multiview data are becoming increasingly useful. In this paper, a novel multiview spectral clustering algorithm is presented (multiview spectral clustering by common eigenvectors, or MVSC-CEV), based on computing the common eigenvectors of the Laplacian matrices derived from the similarity matrices of the input data. This algorithm maintains the features of spectral clustering, while allowing the use of an arbitrary number of input views, possibly of a different nature (feature or graph space) and with different dimensions. The method has been tested on four standard multiview data sets (UCI's Handwritten, BBC segmented news, Max Planck Institute's Animal With Attributes and Reuters multilingual), and compared with seven methods in the state of the art. Seven standard clustering evaluation metrics have been used in the experiments. The quality of the clustering produced by MVSC-CEV is above those obtained by other state-of-the-art methods in the majority of evaluation metrics and dataset combinations. The computation times of this method are approximately twice those of the baseline spectral clustering of the concatenated data views.

Keywords: Multiview data; Spectral clustering; Common eigenvectors

MSC: 41A05; 41A10; 65D05; 65D17

1 Introduction

The development of information and communication technologies has led to ever-increasing data production in most areas of human activity. The difference is not only quantitative, but it is also qualitative, as today it is relatively easy to capture different aspects or features from a given subject or experiment. New pattern recognition methods should be designed, not only to deal with large amounts of information, but also with several modes, feature sets, or views of the same subjects. Social networks provide a good example of multiview or multifeature data. Users of several social networks may have different relationships on each network to which they belong. This produces a different graph of user-to-user relationships for each social network, i.e. several views (relationship graphs) of the same source data (users and the links between them). Biomedicine is another area where multiview data sets are the norm. Given a set of subjects, each has many different information aspects for study, such as genetic sequencing, blood sample analysis, familial relationships to lifestyle, and many others.

Because it requires the development of suitable methods for the analysis and processing of several data views, multiview data presents both challenges and opportunities. In turn, exploiting such data richness may lead to better resolution of pattern recognition problems.

1.1 Multiview clustering interpretation

Given a data set with two or more data views, the goal of multiview clustering is to find a clustering assignment that is compatible with all the input views. Such compatibility has several possible interpretations. In the first interpretation of multiview clustering, two observations *a* and *b* belong to the same cluster only if they belong to the same cluster in *all* the input views, as described for example in [14]; most multiview clustering methods follow this interpretation. A second interpretation extracts all possible clustering assignments from the different views as long as they are orthogonal, i.e. only a single assignment may be extracted (if all views would produce the same clustering assignment), but up to *N* assignments (*N* being the number of views) may exist. This approach is described in [6]. Yet another possible interpretation assigns observations *a* and *b* to the same cluster if they belong to the same cluster in *some* of the input views; the method presented in this paper follows this interpretation.

1.2 Multiview clustering methods

In recent years, several multiview clustering approaches have been developed. In this section, we briefly describe them in order to provide a context for the algorithm presented in this paper.

In general terms, clustering multiple data views $V_1, V_2, \ldots V_c$ involves the following steps:

1. Obtain a similarity matrix S_i for each view V_i.
2. Compute a projection P_i of each S_i into a space suitable for clustering.
3. Produce a clustering assignment.

The main structural difference between the multiview clustering methods proposed in the literature lies in the step where the information from the multiple views is collapsed into a single view to produce the final clustering assignment.

The first category of multiview clustering methods merges the similarity matrices to obtain a combined similarity matrix S' that minimizes the differences between the input similarity matrices S_i (i.e. views are merged in Step 1). Afterwards, a standard clustering algorithm is applied to S' in order to obtain the final clustering. There are several methods in this category, mainly differing in the technique used to compute S', such as Min-disagreement [21], co-training [14], co-regularization [15], feature selection [26] or graph fusion [17].

The second category of multiview clustering methods merge the input views during Step 2 to generate a compatible projection for all views (P'). Afterwards, a standard clustering method is applied to the merged projection P'. An implementation of this approach using Canonical Correlation Analysis in order to maximize the correlation of samples across the projected views can be found in [5].

Ensemble clustering methods are designed to overcome the randomness of clustering methods such as K-means by combining clustering assignments from several runs in order to find a stable assignment. Although not strictly considered as multiview clustering methods, they can be used for multiview clustering if they are applied to the clustering assignments of different views. Thus, they would produce a clustering assignment compatible with all views. These methods merge the information from the different views after Step 3. A survey of ensemble clustering methods can be found in [25], and a more recent method is described in [3].

Finally, the method presented in [27] substantially differs from those above. First, it clusters all views separately. Then, it takes the clustering assignments obtained and loops back to Step 2 (data projection) in order to improve the projections with the clustering information previously obtained. Therefore it is a co-training approach, since it uses the results of one iteration to further improve the results of the final clustering.

The method presented in this paper, as well as an earlier version with a different formulation described in [13], merge the information from the input views during Step 2, as it computes a single projection (the common eigenvector matrix) from the input similarity matrices.

1.3 Goal of the present work

Spectral clustering [22] is a well-known clustering algorithm that is based on spectral graph theory [23]. One of its distinctive features is that it clusters samples by connectivity, not merely by distance, and therefore allows the clustering of data sets with concave or nested clusters.

The goal of this paper is to present a novel multiview spectral clustering algorithm, MVSC-CEV (multiview spectral clustering by common eigenvectors) that extends the features of spectral clustering to multiview scenarios. Our experiments on standard multiview data sets show that the MVSC-CEV algorithm presented here has a better overall performance than existing multiview clustering algorithms.

1.4 Structure of this paper

This paper is organized as follows. Section 2 briefly describes the methods on which the present work is based. Section 3 defines the MVSC-CEV algorithm and presents its theoretical foundations. Section 4 describes the experiments and the multiview data sets used to evaluate the present method. Section 5 presents the results of the experiments on MVSC-CEV and other multiview clustering methods in the state of the art. Finally, Section 6 outlines the main contributions of the work presented in this paper.

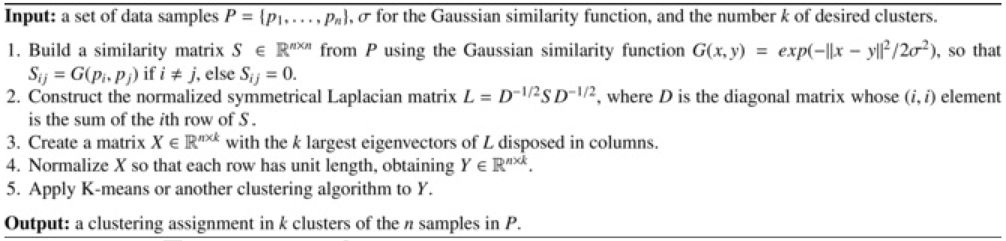
2 Related work

The algorithm described in this paper is mainly based on two well known algorithms: the Ng, Jordan and Weiss (NJW) spectral clustering algorithm [20] and stepwise common principal components method [24]. In this section, we outline both algorithms in order to provide a theoretical background for the MVSC-CEV algorithm described in Section 3.

2.1 Spectral clustering

Spectral graph theory gives the conditions under which a graph can be partitioned into non-connected subgraphs. The spectral clustering algorithm [22] is an application of spectral graph theory to the task of graph clustering, which can be further applied to any data expressed as a matrix of similarities between samples.

There are several variants of the spectral clustering algorithm. The method presented in this paper is based on the variant described in [20], as it is a well accepted and proven variant and it allows a reduction in the computational cost of the MVSC-CEV algorithm, as will be explained in Section 3. As a reference for the definition of MSCV-CEV on Section 3, the NJW spectral clustering algorithm is shown in Algorithm 1.



```
Input: a set of data samples $P = \{p_1, \dots, p_n\}$, $\sigma$ for the Gaussian similarity function, and the number $k$ of desired clusters.
1. Build a similarity matrix $S \in \mathbb{R}^{n \times n}$ from $P$ using the Gaussian similarity function $G(x, y) = exp(-\|x - y\|^2/2\sigma^2)$, so that
   $S_{ij} = G(p_i, p_j)$ if $i \neq j$, else $S_{ij} = 0$.
2. Construct the normalized symmetrical Laplacian matrix $L = D^{-1/2} S D^{-1/2}$, where $D$ is the diagonal matrix whose $(i, i)$ element
   is the sum of the $i$th row of $S$.
3. Create a matrix $X \in \mathbb{R}^{n \times k}$ with the $k$ largest eigenvectors of $L$ disposed in columns.
4. Normalize $X$ so that each row has unit length, obtaining $Y \in \mathbb{R}^{n \times k}$.
5. Apply K-means or another clustering algorithm to $Y$.
Output: a clustering assignment in $k$ clusters of the $n$ samples in $P$.
```

Algorithm 1 Spectral ~~C~~clustering - NJW variant.

alt-text: Algorithm 1

The only parameter besides the desired number of clusters k is σ, which controls the influence of the distance between two points p_i and p_j on their similarity s_{ij}. By default, the heuristic proposed in [18] is used to choose σ. However, the similarity matrix S built in Step 1 can be obtained using a different similarity function.

Note that the algorithm is formulated to receive data in feature space (observations/features matrix) as input. However, it can also be applied to data in graph space (adjacency or similarity matrix).

An important feature of spectral clustering is that it groups points by connectivity, not merely by distance, i.e. if point a is connected to b and b to c then a and c will be assigned to the same cluster, even if the distance between a and c is relatively large. This allows clusterings on data sets with concave groups of points to be found, identifying clusters similarly to how humans identify connected shapes.

2.2 Stepwise common principal components

Common principal components (CPC) analysis, first proposed by Flury [8], is a statistical method of simultaneously diagonalizing a set of positive-definite symmetric matrices. This method, also known as *joint diagonalization*, attempts to diagonalize the input matrices under the hypothesis of common components H, which states that there exists an orthogonal matrix W such that the C input matrices have the same diagonal form, as formulated in Eq. (1).

$$H: L_c' = W^T\ L_c\ W, \quad c = 1, 2, \ldots, C \tag{1}$$

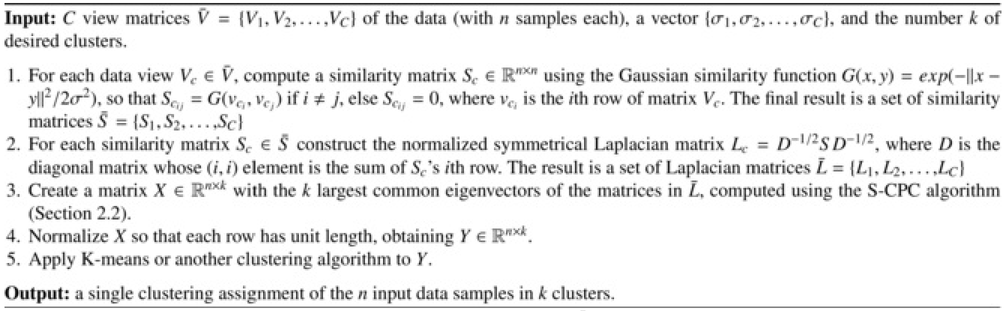
where L_c is the positive-definite symmetric matrix of input matrix c, and L_c' is its diagonalized form, obtained from the linear transformation defined in matrix W. Note that the resulting eigenvectors (columns of W) are common for all the input matrices, while the eigenvalues are specific to each input matrix. In other words, the input matrices are projected into the same subspace defined by the eigenvectors, with the relative weight of each subspace axis for each input matrix given by the associate eigenvalue.

The stepwise common principal components algorithm (referred here as S-CPC), described in [24], finds an approximate solution to this problem in an incremental manner. More specifically, it first computes the common components (common eigenvectors) with highest eigenvalues. Therefore it can stop after computing a given number of common principal components. This has a dramatic impact on the performance of the method presented in this paper, as explained in Section 3. S-CPC can be applied to any number C of input matrices.

3 Method: the MVSC-CEV algorithm

3.1 Description of the algorithm

The Multiview Spectral Clustering algorithm by Common Eigenvectors presented in this paper (MVSC-CEV), detailed in Algorithm 2, follows the same structure than the NJW spectral clustering algorithm described in Algorithm 1. The main difference lies in the fact that MVSC-CEV replaces the single input data matrix with a set of C input data views $\bar{V} = \{V_1, V_2, \ldots, V_C\}$. In turn, for each input matrix in $\bar{V}$, a similarity matrix and its corresponding Laplacian matrix are computed. Then the eigenvectors common to all C Laplacian matrices are computed and used to obtain the clustering.



```
Input: $C$ view matrices $\bar{V} = \{V_1, V_2, \ldots, V_C\}$ of the data (with $n$ samples each), a vector $\{\sigma_1, \sigma_2, \ldots, \sigma_C\}$, and the number $k$ of desired clusters.

1. For each data view $V_c \in \bar{V}$, compute a similarity matrix $S_c \in \mathbb{R}^{n \times n}$ using the Gaussian similarity function $G(x, y) = exp(-\|x - y\|^2/2\sigma^2)$, so that $S_{c_{ij}} = G(v_{c_i}, v_{c_j})$ if $i \neq j$, else $S_{c_{ij}} = 0$, where $v_{c_i}$ is the $i$th row of matrix $V_c$. The final result is a set of similarity matrices $\bar{S} = \{S_1, S_2, \ldots, S_C\}$
2. For each similarity matrix $S_c \in \bar{S}$ construct the normalized symmetrical Laplacian matrix $L_c = D^{-1/2} S D^{-1/2}$, where $D$ is the diagonal matrix whose $(i, i)$ element is the sum of $S_c$'s $i$th row. The result is a set of Laplacian matrices $\bar{L} = \{L_1, L_2, \ldots, L_C\}$
3. Create a matrix $X \in \mathbb{R}^{n \times k}$ with the $k$ largest common eigenvectors of the matrices in $\bar{L}$, computed using the S-CPC algorithm (Section 2.2).
4. Normalize $X$ so that each row has unit length, obtaining $Y \in \mathbb{R}^{n \times k}$.
5. Apply K-means or another clustering algorithm to $Y$.

Output: a single clustering assignment of the $n$ input data samples in $k$ clusters.
```

Algorithm 2 Multiview spectral clustering by common eigenvectors (MVSC-CEV).

alt-text: Algorithm 2

MVSC-CEV can operate on both feature space and graph space input views, and on any combination of both. Input matrices in feature space can have any dimension, possibly varying across the different matrices.

Computational complexity The computational complexity of each step of Algorithm 2 is the following:

1. The cost of the Gaussian similarity function on input view V_c is $O(n^2 m_c)$, where m_c is the number of columns of input view V_c. Thus the cost of computing the Gaussian similarity matrices on all the input views is $O(n^2 M)$, where $M = \sum_{c=1}^{C} m_c$.
2. The total cost of computing the C Laplacian matrices is $O(Cn^2)$.
3. A consequence of using the NJW spectral clustering formulation is that only the k largest common eigenvectors have to be computed. Our variation of S-CPC algorithm makes this possible in an efficient way, thus reducing the computational complexity of Step 3 of the algorithm from $O(Cn^3)$ to $O(Ckn^2)$, with $k \ll n$ in the vast majority of cases.
4. The cost of the normalization step is $O(kn)$.
5. Finally, the cost of K-means is $O(kn^2 + tn^2) = O((k + t)n^2$, where t is the number of iterations K-means is programmed to execute.

Adding the previous terms together gives the overall computational cost, $O((M + C + Ck + k + t)n^2)$.

3.2 Ideal clustering case

To show why Algorithm 2 works as expected, let us consider first an ideal case, where there are $k = 3$ perfectly separated clusters, i.e. the points in different clusters are infinitely far apart from each other. Moreover, all C data views have the same clustering structure. In order to simplify the discussion, let us also assume that the points in the data views are ordered according to the cluster they belong to, so points belonging to cluster 1 appear first, then points of cluster 2 and finally the points of cluster 3.

Consequently, the similarity matrices of this example will be block-diagonal: $\forall S \in \{S_1, S_2, \ldots, S_C\},\ S_{ij} = 0$ if data points i and j do not belong to the same cluster, or greater than zero otherwise. Representing each non-zero subblock of the similarity matrices with a parenthesized superscript:

$$S_c = \begin{bmatrix} S^{(1)} & 0 & 0 \\ 0 & S^{(2)} & 0 \\ 0 & 0 & S^{(3)} \end{bmatrix} \quad \forall S_c \in \bar{S} \tag{2}$$

On the next step of the algorithm, for each similarity matrix S_c a Laplacian matrix L_c is computed, whose block-diagonal structure will be the same:

$$L_c = \begin{bmatrix} L^{(1)} & 0 & 0 \\ 0 & L^{(2)} & 0 \\ 0 & 0 & L^{(3)} \end{bmatrix} \quad \forall L_c \in \bar{L} \tag{3}$$

In step 3 of Algorithm 2, the set of Laplacian matrices $\bar{L}$ defined in (3) is passed to the S-CPC algorithm along with *k* in order to compute their common eigenvectors. According to Trendafilov [24], S-CPC finds the *k* eigenvectors whose sum of eigenvalues is highest. Each such eigenvector is located on the $\mathbb{R}^n$ sphere, where *n* is the number of samples in the input data. The common eigenvectors are mutually orthogonal. Given the common structure of the Laplacian matrices in $L_c \in \bar{L}$, they have the same eigenvectors, which therefore are the common eigenvectors computed by S-CPC. Following Ng et al. [20], the *k* largest eigenvectors of the Laplacian matrices in $L_c \in \bar{L}$ are the first eigenvectors (i.e. those with largest eigenvalue) $x_1^{(i)}$ of each submatrix $L_c^{(i)}$, properly padded with zeros to complete the missing elements:

$$X = \begin{bmatrix} x_1^{(1)} & \vec{0} & \vec{0} \\ \vec{0} & x_1^{(2)} & \vec{0} \\ \vec{0} & \vec{0} & x_1^{(3)} \end{bmatrix} \in \mathbb{R}^{n\times k} \tag{4}$$

Finally, the normalization of the rows of *X* to make them of unit length results in a matrix *Y*, of the form:

$$Y = \begin{bmatrix} y^{(1)} & \vec{0} & \vec{0} \\ \vec{0} & y^{(2)} & \vec{0} \\ \vec{0} & \vec{0} & y^{(3)} \end{bmatrix} \in \mathbb{R}^{n\times k} \tag{5}$$

where $y^{(i)}$ is a vector of ones with as many values as the number of elements in cluster *i*. Applying K-Means to *Y* produces the clustering assignment of the input data samples common to the *C* input views.

3.3 Deviations from the ideal case

On multiview clustering problems there are two possible sources of deviation from the ideal case discussed in Section 3.2.

The first possible situation occurs when the off-diagonal blocks in the similarity matrices $S \in \bar{S}$ are not zero, i.e. the clusters are not perfectly separated from each other. This case is discussed in [20] for a single similarity matrix, but its extension to several similarity matrices with the same structure is straightforward.

The second possible deviation from the ideal case stems from structural discrepancies across data views, where not all similarity matrices share the same structure. Obviously this second deviation implies the former one, as at least some of the views cannot exhibit a perfect block-diagonal structure if there are differences between them.

In this case, the eigenvalues associated to each of the different Laplacian matrices in $\bar{L}$ may not decrease simultaneously; but even in such a case, S-CPC guarantees that the **sum of the eigenvalues** associated to each successive common eigenvector is decreasing. Let $\lambda_c^{(i)}$ be the eigenvalue associated to Laplacian matrix L_c obtained on iteration *i* of S-CPC, i.e. associated with the *i*~~-~~th eigenvector. Therefore, the following relation holds:

$$\sum_{c=1}^{C} \lambda_c^{(i)} \geq \sum_{c=1}^{C} \lambda_c^{(i+1)} \quad \forall i = 1, 2, \ldots, k \tag{6}$$

in other words, the *eigengaps* (difference between consecutive eigenvalues) are conserved:

$$\delta^{(i)} = \sum_{c=1}^{C} \lambda_c^{(i)} - \sum_{c=1}^{C} \lambda_c^{(i+1)} \geq 0 \quad \forall i = 1, 2, \ldots, k \tag{7}$$

This satisfies the matrix perturbation theory condition [23], that guarantees the stability of the *subspace* defined by the first *k* eigenvectors a matrix as long as the eigengaps are conserved.

4 Experimental setup

The MVSC-CEV algorithm is tested on four standard multiview or multifeature data sets in order to validate its operation and compare the quality of the clustering it produces with other state-of-the-art multiview clustering methods. The details of each data set, including the number and size of their data views, as well as the total number of classes and samples, are given in Table 1.

Table 1 Summary of the data sets used in the experiments.

alt-text: Table 1

	Digits			BBC			Reuters			AWA		
View #.	View name	# feat.	σ	View name	# feat.	σ	View name	# feat.	σ	View name	# feat.	σ
1	Pixels	240	0.68476	Segm. A	500/6,838	1.20467	English	500/21,531	0.04631	CQ	~~2,688~~2688	1.470858

2	Fourier coeffs.	76	0.61889	Segm. B	500/6,790	1.20609	French	500/24,892	0.05227	LSS	~~2,000~~2000	0.075466
3	Profile correl.	216	4.19278	—			German	500/34,251	0.05443	PHOG	252	0.929560
4	Zernike coeffs.	47	0.60180	—			Italian	500/15,506	0.05506	SIFT	~~2,000~~2000	0.053216
5	Karhunen moments	64	0.26484	—			Spanish	500/11,547	0.05676	RGSIFT	~~2,000~~2000	0.025925
6	Morph. feats.	6	0.08377	—			—			SURF	~~2,000~~2000	0.091183
# samples	~~2,000~~2000			~~2,112~~2112			~~18758~~18,758			~~30457~~30,457		
# classes	10			5			6			50		

If the number of features column has two numbers, the first is the number of features used and the second the total number of features in the dataset.

The standard Gaussian radial basis function has been used on all four data sets, using the heuristic method proposed in [18] to determine the σ parameter value: to make σ equal to the average distance to the *i*~~-~~th neighbour of each sample, with $i = \log|samples|$. The specific σ values used on each view of the data sets in the experiments are given in Table 1 (columns σ).

4.1 Multiview datasets

4.1.1 Multiple features data set (Digits)

A frequent application of multifeature methods is image data processing, where several feature sets can be extracted from the input images. The University of California at Irvine (UCI) multiple features data set [2], available at the UCI machine learning repository,[1] is created from a set of handwritten numerals (from ~~'~~'0' to '9'), scanned as 15 ~~x~~× 16 pixels images. There are 200 samples of each numeral, resulting in a total of ~~2,000~~2000 samples. The data set provides six different views or feature sets of the original image data.

The input views do not require special preprocessing, and the similarity matrices are computed from the Euclidean distance matrix of each input view.

4.1.2 BBC news (BBC)

The second experiment presented in this paper uses the BBC News multiview text collection [10].[2] ~~.~~ It comprises ~~2,225~~2225 news articles labelled with one of five possible topics (*business, entertainment, politics, sport* or *tech*). The input texts are split into several segments. The term frequencies on each segment become the different input views. There are several subsets in the original data set. In this experiment, the two-segment subset has been chosen to allow direct comparison with the results in the literature. The number of terms in each view, i.e. the number of attributes, is ~~6,838~~6838 and ~~6,790~~6790 respectively, although only the 500 most frequent terms on each segment are used as the less frequent terms do not contribute to the quality of text classification [12]. The *tf.idf* (term frequency / inverse document frequency) [19] is computed on each of the input segments, and the cosine similarity is used instead of the euclidean distance because of the high sparsity of the feature matrices.

4.1.3 Reuters multilingual corpus (Reuters)

The third data set used in the present work is the Reuters multilingual corpus [1],[3] ~~,~~ a set of ~~18758~~18,758 news articles available in five different languages (English, French, German, Italian and Spanish). The subset of original English news articles has been used; the term matrices of the remaining languages come from machine-translated texts. For each input view (language), a matrix with term frequencies is given. As with the *BBC news* data set, only the 500 most frequent terms of each language have been used, their *tf.idf* value has been computed and finally the cosine similarity has been employed to find the similarity matrices.

4.1.4 Animal with attributes (AWA)

The last experiment presented in this paper is the Animal with attributes data set (AWA)[16],[4] which is a multiple feature data set with six standard image features extracted from animal photographs. Again, the Euclidean distance matrix has been computed on the feature matrices and then passed to the Gaussian radial basis function.

4.2 Evaluation metrics

Following the methodology described in [19], we assess the quality of the clustering methods on the four multiview data sets using clustering *purity*, clustering *Rand index* and the *normalized mutual information* (NMI) between clusterings. Also, other supervised clustering quality metrics used are clustering *precision, recall* and their *F-score*, according to the methodology followed in [4,14,27] among others.

In order to evaluate the quality of the partitions in an unsupervised way, the *Dunn index* [7] is used. As the Dunn index measures the quality of a given partition with respect to the original data, in a multiview setting there is a Dunn index value for each input view. The average and standard deviation of the Dunn indices on the original input views are reported, as well as the average Dunn index of the partition defined by the original labels in the dataset as reference value. Dunn index requires the computation of both intra-cluster and inter-cluster distances; for both measures, the average distance has been used, as defined in R package *clv*[5].

Finally, the average execution time of the algorithm is given, along with SC single-view and SC on the concatenated features as reference. The execution time is measured on an Intel i7-6700K @ 4 GHz with 32GB of RAM, using single-threaded processes.

5 Evaluation and results

The clustering quality of MVSC-CEV is evaluated with respect to both baseline clustering methods and multiview clustering methods in the state of the art. As baseline method, spectral clustering [20] (**SC view #**) is applied separately on each input view. Also, spectral clustering is applied to the concatenation of all input views into a single matrix (**SC concat.**) to appraise the difference of having the same information on separate views or on a single view. On the other hand, the method presented here is compared with the reported results of seven highly relevant state-of-the-art multiview clustering methods: co-training multiview spectral clustering [14] (**CotrainSC**), multi-modal spectral clustering [4] (**MMSC**), multiview clustering via structured sparsity [26] (**MVC-SS**), multiview K-means clustering [3] (**MV-KMeans**), multiview clustering via joint nonnegative matrix factorization [9] (**MVC-JNMF**), subspace co-training for multiview clustering [27] (**CoKmLDA**), and large-scale multiview spectral clustering via bipartite graph [17] (**MVSC-BG**).

Table 2 shows the results of the supervised clustering metrics on the different clustering methods applied to the handwritten digits dataset. The results on single view SC configurations show certain differences among views, while the SC with concatenated features presents clearly better results, showing that the incorporation of the information from the different views is useful on this dataset. Regarding the multiview methods in the state of the art, in general their performance is below SC on concatenated features. However, MVSC-CEV produces the highest results in all six metrics, showing a better exploitation of the multiview data than the other methods.

Table 2 Clustering performance on the handwritten digits dataset.

alt-text: Table 2

Method	F-score	Precision	Recall	NMI	Rand ~~I~~index	Purity
SC view 1	0.615	0.611	0.620	0.671	0.568	0.721
SC view 2	0.532	0.466	0.619	0.632	0.467	0.651
SC view 3	0.587	0.573	0.602	0.660	0.536	0.697
SC view 4	0.644	0.621	0.669	0.709	0.599	0.750
SC view 5	0.495	0.462	0.533	0.562	0.430	0.610
SC view 6	0.634	0.597	0.677	0.685	0.587	0.744
SC concat.	0.837	0.833	0.841	0.851	0.817	0.910
CotrainSC	0.726	0.709	0.745	0.765	0.695	n/a
MMSC	n/a	n/a	n/a	0.792	n/a	0.758
MV-KMeans	n/a	n/a	n/a	0.807	0.789	0.825
MVC-JNMF	n/a	n/a	n/a	0.804	0.881	n/a
CoKmLDA	n/a	n/a	n/a	0.818	0.819	n/a
MVSC-BG	n/a	n/a	n/a	0.832	n/a	0.844
MVSC-CEV	**0.899**	**0.897**	**0.900**	**0.892**	**0.886**	**0.946**

n/a means result not available.

The BBC segmented news dataset is a very homogeneous dataset, as both of its views have the same structure (bag of words) and in fact share most of the features. Therefore the results on this dataset, shown in Table 3, are expected to be quite similar across the different configurations. The single-view SC configurations show very similar results. The SC concatenated configuration performs slightly better, probably due to the addition of new features (words) with respect to the single views. As for the multiview methods, only two papers report results for this dataset. CotrainSC shows the best results on the different metrics when applied to this dataset. However, these are reported results from [14] for CotrainSC, where the preprocessing is quite different for this dataset, as it uses the probabilistic latent semantic analysis matrices [11] as input to CotrainSC, so these results have to be compared with caution. MVSC-CEV results are slightly below those of CotrainSC, although they are consistently above the results of SC concat. In general, the numerical differences in these results are small, probably as a consequence of the aforementioned homogeneity of the views of this dataset.

Table 3 Clustering performance on the BBC segmented news dataset.

alt-text: Table 3

Method	F-score	Precision	Recall	NMI	Rand ~~I~~index	Purity
SC view 1	0.852	0.850	0.854	0.775	0.813	0.921
SC view 2	0.847	0.845	0.848	0.771	0.807	0.917
SC concat.	0.871	0.867	0.875	0.810	0.837	0.932
CotrainSC	**0.898**	**0.894**	**0.902**	**0.841**	**0.873**	n/a
CoKmLDA	n/a	n/a	n/a	0.796	n/a	0.914

MVSC-CEV	0.884	0.884	0.885	0.826	0.821	**0.940**

n/a means result not available.

Single-view SC on the Reuters multilingual news dataset, in Table 4, shows differences between the different views, i.e. languages of the source documents. Some input languages tend to perform worse than others, although the trends are not shared across the different evaluation metrics. In fact, SC on the concatenated features tends to perform better than its single-view counterparts, but that does not hold on all metrics. Apparently this implies that this is a complex dataset, where the simple addition of features does not guarantee an improvement in the results. Some multiview methods fail to improve the results of SC or SC concatenated, like MMSC and MVC-SS. Overall the best results are achieved by MVSC-CEV, except for the NMI value where MVC-JNMF produces the highest result.

Table 4 Clustering performance on the Reuters multilingual dataset.

alt-text: Table 4

Method	F-score	Precision	Recall	NMI	Rand ~~I~~index	Purity
SC view 1	0.365	0.315	0.435	0.310	0.154	0.531
SC view 2	0.288	0.219	0.421	0.303	0.110	0.549
SC view 3	0.376	0.340	0.420	0.304	0.181	0.569
SC view 4	0.296	0.216	0.471	0.312	0.201	0.538
SC view 5	0.336	0.270	0.445	0.308	0.193	0.568
SC concat.	0.379	0.316	0.473	0.342	0.162	0.550
CotrainSC	0.412	0.369	0.467	0.388	0.279	n/a
MMSC	n/a	n/a	n/a	0.134	n/a	0.390
MVC-SS	n/a	n/a	n/a	n/a	n/a	0.531
MVC-JNMF	n/a	n/a	n/a	**0.409**	**0.535**	n/a
MVSC-BG	n/a	n/a	n/a	0.357	n/a	0.577
MVSC-CEV	**0.459**	**0.394**	**0.550**	0.341	**0.536**	**0.619**

n/a means result not available.

The results on the animal with attributes dataset, presented in Table 5, show a high variability between single views, where some of the clusterings on single views have the highest indicators on some metrics: spectral clustering on view 4 (SIFT) achieves the highest F-score and precision of all the methods evaluated. Probably due to these differences among views, the SC of concatenated features performs below most of the single-view clusterings. Regarding the multiview methods, there is also a high variability, with MVSC-CEV obtaining the highest recall, NMI, Rand Index and purity values.

Table 5 Clustering performance on the animal with attributes (AWA) dataset.

alt-text: Table 5

Method	F-score	Precision	Recall	NMI	Rand ~~I~~index	Purity
SC view 1	0.162	0.124	0.233	0.359	0.544	0.273
SC view 2	0.485	0.455	0.519	0.742	0.774	0.593
SC view 3	0.124	0.108	0.145	0.291	0.552	0.235
SC view 4	**0.550**	**0.521**	0.583	0.796	0.778	0.647
SC view 5	0.353	0.282	0.473	0.644	0.659	0.439
SC view 6	0.451	0.374	0.567	0.743	0.768	0.566
SC concat.	0.314	0.280	0.358	0.553	0.563	0.460
MMSC	n/a	n/a	n/a	0.698	0.585	n/a

MVC-SS	n/a	n/a	n/a	0.751	0.629	n/a
MV-KMeans	n/a	n/a	n/a	0.117	0.094	0.114
MVSC-CEV	0.545	0.493	**0.609**	**0.833**	**0.876**	**0.795**

n/a means result not available.

The Dunn index results, reported in Table 6, are higher for single-view spectral clustering. This is expected as the single-view SC clustering produces a better clustering for a specific input view, i.e. it is more specialized. On the other hand, both SC concat. and MVSC-CEV have to produce a clustering that is compatible with all the input views, therefore obtaining lower Dunn indices on some of the views as they cannot specialize on a single view. In any case, the reference clustering (the groundtruth labels in the dataset) produces Dunn indices below both the best single-view and MVSC-CEV in three datasets, suggesting that the reference clustering does not match the underlying structure of the data as well as the clusterings produced by the different methods.

Table 6 Dunn index on the four data sets.

alt-text: Table 6

Method	Digits	BBC	Reuters	AWA
Reference	0.823 ± 0.267	0.996 ± 0.006	0.625 ± 0.018	0.525 ± 0.117
SC view 1	0.770 ± 0.200	1.003 ± 0.009	0.498 ± 0.016	0.312 ± 0.188
SC view 2	0.797 ± 0.213	**1.006 ± 0.006**	0.771 ± 0.176	0.482 ± 0.146
SC view 3	0.932 ± 0.169	—	0.684 ± 0.029	0.443 ± 0.110
SC view 4	**0.977 ± 0.179**	—	**0.868 ± 0.126**	0.363 ± 0.229
SC view 5	0.747 ± 0.190	—	0.665 ± 0.030	0.415 ± 0.124
SC view 6	0.804 ± 0.146	—	—	**0.505 ± 0.157**
SC concat	0.902 ± 0.160	0.999 ± 0.006	0.743 ± 0.050	0.428 ± 0.136
MVSC-CEV	0.846 ± 0.202	1.003 ± 0.007	0.826 ± 0.049	0.295 ± 0.143

Finally, Table 7 shows the average computation time over ten executions of standard spectral clustering on single and concatenated views, as well as the computation time of MVSC-CEV. As expected, single-view clustering is the fastest method, as it receives less input data. Spectral clustering on the concatenated views has a clearly higher cost, specially on larger datasets, and MVSC-CEV has an even higher cost, around twice the cost of SC concat., due to the added complexity of finding the common eigenvectors.

Table 7 Execution times (seconds) on the four data sets.

alt-text: Table 7

Method	Digits	BBC	Reuters	AWA
SC single-view	21.3 ± 1.4	10.5 ± 0.0	891.8 ± 46.0	1331.3 ± 126.9
SC concat.	27.6 ± 1.4	16.5 ± 0.8	2017.2 ± 97.4	3886.7 ± 415.2
MVSC-CEV	77.5 ± 2.6	18.2 ± 0.7	3590.3 ± 107.5	8167.5 ± 649.3

6 Conclusions

This paper presents MVSC-CEV, a novel multiview spectral clustering algorithm that finds a clustering common to all the input views of a multiview dataset by computing the common eigenvectors of the Laplacian matrices. It extends the properties of spectral clustering to multiview scenarios, specifically the connectivity-based (as opposed to distance-based) clustering of the data, as well as giving the possibility of working with input data in either feature space or graph space. In fact, MVSC-CEV can work with any combination of views in feature or graph space, and the dimension of the feature sets in the former case is arbitrary and can be different from view to view. The algorithm has been tested on four standard multiview datasets

The algorithm has shown overall better clustering quality on three standard benchmark multiview datasets out of four datasets tested, measured with six clustering evaluation metrics. It has been compared with seven state-of-the-art multiview clustering methods. Even though CotrainSC [14] has achieved slightly better results on the BBC segmented news dataset, this is a very homogeneous dataset, as both views are strongly correlated, and using multiview methods on it does not improve greatly the quality of the results. However, on the other datasets, where the input views are more heterogeneous, MVSC-CEV in general ranks better than the other methods and also better than single-view baseline methods.

The quality of the clustering as estimated by the Dunn index has shown somehow contradictory results, as single-view clusterings have higher Dunn indices, probably due to their specialization in finding the most adequate clustering for a specific view. MVSC-CEV obtains slightly lower Dunn indices, although the reference class labeling in the dataset obtains even lower Dunn indices in three datasets. This seems to indicate that the underlying data structure does not follow the class assignment very closely.

Finally, the implementation of the algorithm is computationally efficient, because only *k* eigenvectors need to be computed in order to produce a clustering assignment with *k* clusters, thus dramatically reducing the computation time of the common eigenvector projection. In fact, the computation times on the datasets are approximately twice those of spectral clustering with concatenated data views, which uses the same amount of input information. In exchange, MVSC-CEV produces clearly better clusterings on almost all dataset and metric combinations.

Acknowledgements

This work was supported by grants from the TEC2013-44666-R, the TEC2014-60337-R and the (2009SGR-1395) consolidated research group of the Generalitat de Catalunya, Spain. CIBER-BBN is an initiative of the Spanish ISCIII.

References

[1] M. Amini, N. Usunier and C. Goutte, Learning from multiple partially observed views – an application to multilingual text categorization, *Adv. Neural Inf. Process. Syst.* **22**, 2009, 28–36.

[2] M. Breukelen, R.P.W. Duin, D.M.J. Tax and J.E. Hartog, Handwritten digit recognition by combined classifiers, *Kybernetika* **34** (4), 1998, [381]–386.

[3] X. Cai, F. Nie and H. Huang, Multi-view K-means clustering on big data, IJCAI International Joint Conference on Artificial Intelligence, 2013, 2598–2604.

[4] X. Cai, F. Nie, H. Huang and F. Kamangar, Heterogeneous image feature integration via multi-modal spectral clustering, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2011, 1977–1984.

[5] K. Chaudhuri, S.M. Kakade, K. Livescu and K. Sridharan, Multi-view clustering via canonical correlation analysis, Proceedings of the 26th Annual International Conference on Machine Learning, *ICML '09* 2009, ACM, New York, NY, USA, 129–136.

[6] Y. Cui, X.Z. Fern and J.G. Dy, Non-redundant multi-view clustering via orthogonalization, Seventh IEEE International Conference on Data Mining, *ICDM 2007* 2007, 133–142.

[7] J.C. Dunn, Well-separated clusters and optimal fuzzy partitions, *J. Cybern.* **4** (1), 1974, 95–104.

[8] B.N. Flury, Common principal components in k groups, *J. Am. Stat. Assoc.* **79** (388), 1984, 892–898.

[9] J. Gao, J. Han, J. Liu and C. Wang, Multi-view clustering via joint nonnegative matrix factorization., SDM, 2013, SIAM, 252–260.

[10] D. Greene and P. Cunningham, A matrix factorization approach for integrating multiple data views, Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part I, *ECML PKDD '09* 2009, Springer-Verlag, Berlin, Heidelberg, 423–438.

[11] T. Hofmann, Probabilistic latent semantic indexing, Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1999, ACM, 50–57.

[12] T. Joachims, A statistical learning learning model of text classification for support vector machines, Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, 2001, 128–136.

[13] S. Kanaan-Izquierdo, A. Ziyatdinov, R. Massanet and A. Perera, Multiview approach to spectral clustering, 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2012, 1254–1257.

[14] A. Kumar and H. Daumé III, A co-training approach for multi-view spectral clustering, International Conference on Machine Learning (ICML), 2011, 393–400.

[15] A. Kumar, P. Rai and H. Daumé III, Co-regularized multi-view spectral clustering, Neural Information Processing Systems (NIPS), 2011, 1413–1421.

[16]
C.H. Lampert, H. Nickisch and S. Harmeling, Learning to detect unseen object classes by between-class attribute transfer, 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009, 2009, 951–958.

[17]
Y. Li, F. Nie, H. Huang and J. Huang, Large-scale multi-view spectral clustering with bipartite graph, Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.

[18]
U. Luxburg, A tutorial on spectral clustering, *Stat. Comput.* **17** (4), 2007, 395–416.

[19]
C.D. Manning, P. Raghavan and H. Schütze, Introduction to information retrieval, *J. Am. Soc. Inf. Sci. Technol.* **1**, 2008, 496.

[20]
A.Y. Ng, M.I. Jordan and Y. Weiss, On spectral clustering: analysis and an algorithm, *Adv. Neural Inf. Process. Syst.* 2001, 849–856.

[21]
V.R. de Sa, Spectral clustering with two views, ICML (International Conference on Machine Learning) Workshop on Learning with Multiple Views, 2005, 20–27.

[22]
J. Shi and J. Malik, Normalized cuts and image segmentation, *Pattern Anal. Mach. Intell. IEEE Trans.* **22** (8), 2000, 888–905.

[23]
G.W. Stewart and H.B. Jovanovich, Matrix perturbation theory, *Math. Comput. Simul.* **33** (1), 1991, 74.

[24]
N.T. Trendafilov, Stepwise estimation of common principal components, *Comput. Stat. Data Anal.* **54** (12), 2010, 3446–3457.

[25]
Vega-Pons and Ruiz-Shulcloper, A survey of clustering ensemble algorithms, *Int. J. Pattern Recognit Artif Intell.* **25** (03), 2011, 337–372.

[26]
H. Wang, F. Nie and H. Huang, Multi-view clustering and feature learning via structured sparsity, Proceedings of the 30th International Conference on Machine Learning (ICML-13), **28**, 2013, 352–360.

[27]
X. Zhao, N. Evans and J.-L. Dugelay, A subspace co-training framework for multi-view clustering, *Pattern Recognit. Lett.* **41** (0), 2014, 73–82.

Footnotes

[1] https://www.archive.ics.uci.edu/ml/datasets/Multiple+Features (Accessed May-2017).

[2] http://www.mlg.ucd.ie/datasets/bbc.html (Accessed May-2017).

[3] https://www.archive.ics.uci.edu/ml/datasets/Reuters+RCV1+RCV2+Multilingual+Multiview+Text+Categorization+Test+collection (Accessed May-2017).

[4] http://www.attributes.kyb.tuebingen.mpg.de/ (Accessed May-2017).

[5] https://www.cran.r-project.org/package=clv (Accessed June-2017).

Highlights

- Multiview spectral clustering algorithm.
- Computes the common eigenvectors of the data similarity Laplacian matrices.
- Generic: any number, size and kind of views (both feature and graph space).
- Better overall performance than existing methods, tested on standard multiview data sets.
- Optimized method: to find *k* clusters only needs to compute *k* common eigenvectors.

Queries and Answers

elsevier_PATREC_7027

Query: Please confirm that givennames and surnames have been identified correctly.
Answer: Yes