Information Fusion 44 (2018) 46–56

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Multi-View Kernel Spectral Clustering

Lynn Houthuys*, Rocco Langone, Johan A.K. Suykens

Department of Electrical Engineering ESAT-STADIUS, KU Leuven Kasteelpark Arenberg 10 B-3001 Leuven, Belgium

ARTICLE INFO

Keywords:
Multi-view learning
Clustering
Out-of-sample extension
Kernel CCA

ABSTRACT

In multi-view clustering, datasets are comprised of different representations of the data, or views. Although each view could individually be used, exploiting information from all views together could improve the cluster quality. In this paper a new model Multi-View Kernel Spectral Clustering (MVKSC) is proposed that performs clustering when two or more views are available. This model is formulated as a weighted kernel canonical correlation analysis in a primal-dual optimization setting typical of Least Squares Support Vector Machines (LS-SVM). The primal model includes, in particular, a coupling term, which enforces the clustering scores corresponding to the different views to align. Because of the out-of-sample extension, this model is easily applied to large-scale datasets. The performance of the proposed model is shown on synthetic and real-world datasets, as well as on some large-scale datasets. Experimental comparisons with a number of other methods show that using multiple views improves the clustering results and that the proposed method is competitive with other state-of-the-art algorithms in terms of clustering accuracy and runtime. Especially on the large-scale datasets the advantage of the proposed method is clearly shown, as it is able to handle larger datasets than the other state-of-the-art algorithms.

1. Introduction

In various application domains, data from different sources or *views* are available. Many real-world datasets have representations in the form of multiple views [1]. For example, web pages can be classified based on both the page content (text) and hyperlink information [2], for social networks one could use the user profile but also the friend links [3], images can be classified based on the colors as well as the texture [4], and so on. Although each of the views by itself might already be sufficient for a given learning task, additional views often provide complementary information which can lead to an improved performance [5]. For an extensive overview of recent multi-view learning methods we refer to the work of Zhao et al. [6].

The information from multiple views can be fused in different ways as well as in different stages of the training process. In early fusion techniques, the views are fused before the training process starts, e.g. by means of feature concatenation [7] or in a more complex way like the work done by e.g. Yu et al. [8] and Lin et al. [9]. In this way the information from all views is taken into account early on in the training process. In late fusion techniques the models are usually trained separately and a combination of the individual results is taken to determine the final result. This combination can be formed in many ways, like for example by taking a weighted average, e.g. as done by Bekker et al. [10] for classification, or selective voting, e.g. as done by Xie et al. [11] for clustering.

The *clustering* problem [12] refers to the task of finding a partition of a given dataset based on some similarity measure between the examples. While there are various clustering algorithms available (e.g. the work by Sharma et al. [13,14] and Elhamifar et al. [15]), Spectral Clustering methods are increasingly popular due to the well-defined mathematical framework and its strong performance on arbitrary shaped clusters [16]. Spectral clustering methods make use of the eigenvectors of a rescaled affinity matrix derived from the data (i.e. the Laplacian) to divide a dataset into natural groups, such that points within the same group are similar and points in different groups are dissimilar to each other [17–19]. *Kernel Spectral Clustering* (KSC) [20] is a well-known clustering technique that represents a spectral clustering formulation as a weighted kernel PCA problem, cast in the LS-SVM framework [21].

In this paper a new model is introduced, called *Multi-View Kernel Spectral Clustering* (MVKSC)[1], which is an extension to KSC that allows to deal with multiple data-sources. This is done by integrating two or more KSC models in the joint MVKSC approach and adding a coupling term which maximizes the correlation of the score variables. This coupling can be thought of as a combination of early and late fusion, where the information of all views is already exploited during the

* Corresponding author.
E-mail addresses: lynn.houthuys@esat.kuleuven.be (L. Houthuys), rocco.langone@esat.kuleuven.be (R. Langone), johan.suykens@esat.kuleuven.be (J.A.K. Suykens).

[1] The Matlab implementation of the MVKSC algorithm is available for downloading from http://www.esat.kuleuven.be/stadius/ADB/software.php.

https://doi.org/10.1016/j.inffus.2017.12.002
Received 20 April 2017; Received in revised form 20 September 2017; Accepted 16 December 2017
Available online 18 December 2017

training phase while still allowing for some degree of freedom to model the data from the different views differently.

Furthermore, the proposed model is also closely related to *Kernel Canonical Correlation Analysis* (KCCA) [21], which is a method for determining nonlinear relations among several variables. Although the KCCA learning task is essentially different from clustering, the two formulations are similar.

Expanding spectral clustering techniques to multi-view learning has been done in the past, for example by Cai et al. [22], Kumar et al. [23], Xie et al. [11] and Xia et al. [24]. Although these methods have achieved good accuracy, they are usually computationally expensive and not suitable for large-scale data. Li et al. [25] designed a method to deal with large-scale data by forming a bipartite graph for each view and running spectral clustering on the fusion of all graphs.

Similar to KSC, MVKSC has a natural out-of-sample extension to deal with new test data. Due to this extension the method is able to deal with large-scale data by training on only a small randomly chosen subset. This approach was used for KSC on large-scale network data by Mall et al. [26], although the authors did not simply pick the subset at random but used an algorithm that preserves the overall community structure. There are more complex extensions to KSC to deal with large-scale data, for example the fixed-size approach done by Langone & Suykens [27], but we show here that even this simple approach achieves good performance.

This paper shows how the clustering performance achieved by KSC on one view can be improved by exploiting information from multiple different views. The paper further shows that the out-of-sample extension can be used to deal with large-scale data in a natural way and shows the performance of MVKSC on a real-world large-scale dataset.

We will denote matrices as bold uppercase letters and vectors as bold lowercase letters. The superscript $^{[v]}$ will denote the vth set of variables for KCCA or the vth view for the multi-view method. Whereas the superscript $^{(l)}$ will denote the lth binary clustering problem in case there are more than two clusters.

The rest of this paper is organized as follows: Section 2.1 and Section 2.2 give a summary of the KCCA and the KSC model respectively. Section 3 discusses the proposed model MVKSC. It shows the mathematical formulation, explains the cluster assignment for the training data as well as for the out-of-sample test data and describes the model selection process. Section 4 discusses the experiments done with MVKSC and compares it to other state-of-the-art methods, and to KSC on the separate views alone. Section 4 further discusses the obtained results. Section 5 shows the performance of MVKSC when handling large-scale data. Finally, in Section 6 some conclusions are drawn.

2. Background

This section introduces the concepts of Kernel Canonical Correlation Analysis (KCCA) and Kernel Spectral Clustering (KSC).

2.1. Kernel Canonical Correlation Analysis

Canonical Correlation Analysis (CCA) was originally studied by Hotelling [28] and is a statistical method for determining linear relations among several variables. A nonlinear extension of CCA was introduced by Lai and Fyfe [29], Bach and Jordan [30] and by Van Gestel et al. [31] as *kernel CCA* or KCCA. To determine nonlinear relations, the input space is mapped to a high-dimensional feature space where classical CCA is applied.

A formulation in the LS-SVM framework was proposed by Suykens et al. [21]. Given data $\mathscr{D}^{[1]} = \{\mathbf{x}_i^{[1]}\}_{i=1}^N \subset \mathbb{R}^{d^{[1]}}$ and $\mathscr{D}^{[2]} = \{\mathbf{x}_i^{[2]}\}_{i=1}^N \subset \mathbb{R}^{d^{[2]}}$, the primal model of KCCA is formulated as follows:

$$\max_{\substack{\mathbf{w}^{[1]},\mathbf{w}^{[2]},\\ \mathbf{e}^{[1]},\mathbf{e}^{[2]}}} -\frac{1}{2}\mathbf{w}^{[1]^T}\mathbf{w}^{[1]} - \frac{1}{2}\mathbf{w}^{[2]^T}\mathbf{w}^{[2]} - \gamma^{[1]}\frac{1}{2}\mathbf{e}^{[1]^T}\mathbf{e}^{[1]} - \gamma^{[2]}\frac{1}{2}\mathbf{e}^{[2]^T}\mathbf{e}^{[2]} + \rho\mathbf{e}^{[1]^T}\mathbf{e}^{[2]}$$

$$\begin{aligned} \text{s.t.} \quad & \mathbf{e}^{[1]} = (\boldsymbol{\Phi}^{[1]} - \mathbf{1}_N\hat{\boldsymbol{\mu}}^{[1]^T})\mathbf{w}^{[1]}, \\ & \mathbf{e}^{[2]} = (\boldsymbol{\Phi}^{[2]} - \mathbf{1}_N\hat{\boldsymbol{\mu}}^{[2]^T})\mathbf{w}^{[2]} \end{aligned} \tag{1}$$

where $\mathbf{e}^{[1]} \in \mathbb{R}^N$ and $\mathbf{e}^{[2]} \in \mathbb{R}^N$ are the score variables indicating the nonlinear relations. $\boldsymbol{\Phi}^{[1]} \in \mathbb{R}^{N\times d_h^{[1]}}$ and $\boldsymbol{\Phi}^{[2]} \in \mathbb{R}^{N\times d_h^{[2]}}$ are feature matrices with $\boldsymbol{\Phi}^{[1]} = [\varphi^{[1]}(\mathbf{x}_1^{[1]})^T;\cdots;\varphi^{[1]}(\mathbf{x}_N^{[1]})^T]$ and $\boldsymbol{\Phi}^{[2]} = [\varphi^{[2]}(\mathbf{x}_1^{[2]})^T;\cdots;\varphi^{[2]}(\mathbf{x}_N^{[2]})^T]$ where $\varphi^{[1]}: \mathbb{R}^{d^{[1]}} \to \mathbb{R}^{d_h^{[1]}}$ and $\varphi^{[2]}: \mathbb{R}^{d^{[2]}} \to \mathbb{R}^{d_h^{[2]}}$ are the mappings to high-dimensional feature spaces. $\hat{\mu}^{[1]} = (1/N)\sum_{i=1}^N\varphi^{[1]}(\mathbf{x}_i^{[1]}) = (1/N)\boldsymbol{\Phi}^{[1]^T}\mathbf{1}_N$ and $\hat{\boldsymbol{\mu}}^{[2]} = (1/N)\sum_{i=1}^N \varphi^{[2]}(\mathbf{x}_i^{[2]}) = (1/N)\boldsymbol{\Phi}^{[2]^T}\mathbf{1}_N$ are used to center the data and $\gamma^{[1]} \in \mathbb{R}^+$ and $\gamma^{[2]} \in \mathbb{R}^+$ are regularization parameters.

The dual problem related to this primal formulation is:

$$\begin{bmatrix} \mathbf{0}_N & \boldsymbol{\Omega_c}^{[2]} \\ \boldsymbol{\Omega_c}^{[1]} & \mathbf{0}_N \end{bmatrix}\begin{bmatrix} \alpha^{[1]} \\ \alpha^{[2]} \end{bmatrix} = \frac{1}{\rho}\begin{bmatrix} \gamma^{[1]}\boldsymbol{\Omega_c}^{[1]} + I & \mathbf{0}_N \\ \mathbf{0}_N & \gamma^{[2]}\boldsymbol{\Omega_c}^{[2]} + I \end{bmatrix}\begin{bmatrix} \alpha^{[1]} \\ \alpha^{[2]} \end{bmatrix} \tag{2}$$

where $\boldsymbol{\Omega_c}^{[1]} = (\boldsymbol{\Phi}^{[1]} - \mathbf{1}_N\hat{\boldsymbol{\mu}}^{[1]^T})(\boldsymbol{\Phi}^{[1]} - \mathbf{1}_N\hat{\boldsymbol{\mu}}^{[1]^T})^T$ and $\boldsymbol{\Omega_c}^{[2]} = (\boldsymbol{\Phi}^{[2]} - \mathbf{1}_N\hat{\boldsymbol{\mu}}^{[2]^T})(\boldsymbol{\Phi}^{[2]} - \mathbf{1}_N\hat{\boldsymbol{\mu}}^{[2]^T})^T$ are the centered kernel matrices and where

$$\begin{aligned} \Omega_{c_{kl}}^{[1]} &= (\varphi^{[1]}(\mathbf{x}_k^{[1]}) - \hat{\boldsymbol{\mu}}^{[1]})^T(\varphi^{[1]}(\mathbf{x}_l^{[1]}) - \hat{\boldsymbol{\mu}}^{[1]}) \\ \Omega_{c_{kl}}^{[2]} &= (\varphi^{[2]}(\mathbf{x}_k^{[2]}) - \hat{\boldsymbol{\mu}}^{[2]})^T(\varphi^{[2]}(\mathbf{x}_l^{[2]}) - \hat{\boldsymbol{\mu}}^{[2]}) \end{aligned} \tag{3}$$

are the elements of these centered kernel matrices for $k, l = 1, \ldots, N$. In practice they can be computed by $\boldsymbol{\Omega_c}^{[1]} = \mathbf{M_c}\boldsymbol{\Omega}^{[1]}\mathbf{M_c}$ and $\boldsymbol{\Omega_c}^{[2]} = \mathbf{M_c}\boldsymbol{\Omega}^{[2]}\mathbf{M_c}$ where $\mathbf{\Omega}^{[1]}$ and $\mathbf{\Omega}^{[2]}$ are the kernel matrices with $\boldsymbol{\Omega}^{[1]} = \boldsymbol{\Phi}^{[1]}\boldsymbol{\Phi}^{[1]^T}$ and $\boldsymbol{\Omega}^{[2]} = \boldsymbol{\Phi}^{[2]}\boldsymbol{\Phi}^{[2]^T}$ and where $\Omega_{ij}^{[1]} = K^{[1]}(\mathbf{x}_i^{[1]}, \mathbf{x}_j^{[1]}) = \varphi^{[1]}(\mathbf{x}_i^{[1]})^T\varphi^{[1]}(\mathbf{x}_j^{[1]})$ and $\Omega_{ij}^{[2]} = K^{[2]}(\mathbf{x}_i^{[2]}, \mathbf{x}_j^{[2]}) = \varphi^{[2]}(\mathbf{x}_i^{[2]})^T\varphi^{[2]}(\mathbf{x}_j^{[2]})$ and $\mathbf{M_c} = \mathbf{I}_N - (1/N)\mathbf{1}_N\mathbf{1}_N^T$ is a centering matrix. $\boldsymbol{\alpha}^{[1]}$ and $\boldsymbol{\alpha}^{[2]}$ are the Lagrange multipliers relayed to the constraints in Eq. (1), also called the dual variables. The kernel functions $K^{[1]}: \mathbb{R}^{d^{[1]}} \times \mathbb{R}^{d^{[1]}} \to \mathbb{R}$ and $K^{[2]}: \mathbb{R}^{d^{[2]}} \times \mathbb{R}^{d^{[2]}} \to \mathbb{R}$ are similarity functions and have to be positive definite.

The eigenvalues and eigenvectors that give an optimal correlation coefficient value are selected. The score variables on the training data can be computed by:

$$\begin{aligned} \mathbf{e}^{[1]} &= \boldsymbol{\Omega_c}^{[1]}\boldsymbol{\alpha}^{[1]} \\ \mathbf{e}^{[2]} &= \boldsymbol{\Omega_c}^{[2]}\boldsymbol{\alpha}^{[2]}. \end{aligned} \tag{4}$$

Since the KCCA method is used to find interesting relations between variables it could be applied to do input selection. It is however important to make a good choice of the regularization constants $\gamma^{[1]}$ and $\gamma^{[2]}$ and of the kernels and their tuning parameters. For this purpose an additional validation set can be used to ensure meaningful generalization of the method.

2.2. Kernel Spectral Clustering

This section summarizes the Kernel Spectral Clustering (KSC) model as introduced by Alzate & Suykens [20]. KSC represents a spectral clustering formulation as a weighted kernel PCA problem, cast in the LS-SVM framework [21].

Given training data $\mathscr{D} = \{\mathbf{x}_i\}_{i=1}^N \subset \mathbb{R}^d$ and the number of clusters k, the primal model of KSC is formulated as follows:

$$\begin{aligned} \min_{\mathbf{w}^{(l)},\mathbf{e}^{(l)},b^{(l)}} \quad & \frac{1}{2}\sum_{l=1}^{k-1}\mathbf{w}^{(l)T}\mathbf{w}^{(l)} - \frac{1}{2N}\sum_{l=1}^{k-1}\gamma^{(l)}\mathbf{e}^{(l)T}\mathbf{D}^{-1}\mathbf{e}^{(l)} \\ \text{s.t.} \quad & \mathbf{e}^{(l)} = \boldsymbol{\Phi}\boldsymbol{w}^{(l)} + b^{(l)}\mathbf{1}_N,\ l = 1,\cdots,k-1 \end{aligned} \tag{5}$$

where $\mathbf{e}^{(l)} = [e_1^{(l)}, \ldots, e_N^{(l)}]^T$ are the clustering scores or projections, $l = 1, \ldots, k-1$ indicate the score variables needed to encode k clusters, $\boldsymbol{\Phi} \in \mathbb{R}^{N\times d_h}$ is the feature matrix with $\boldsymbol{\Phi} = [\varphi(\mathbf{x}_1)^T;\cdots;\varphi(\mathbf{x}_N)^T]$ where $\varphi: \mathbb{R}^d \to \mathbb{R}^{d_h}$ is the mapping to a high-dimensional feature space, $b^{(l)}$ are bias terms, $\mathbf{D}^{-1} \in \mathbb{R}^{N\times N}$ is the inverse of the degree matrix $\mathbf{D}$ with