Clustering From Two Data Sources Using a Kernel-Based Approach with Weight Coupling

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

Department of Electrical Engineering ESAT-STADIUS, KU Leuven
Kasteelpark Arenberg 10 B-3001 Leuven, Belgium
Email: {lynn.houthuys, rocco.langone, johan.suykens}@esat.kuleuven.be

Abstract. In many clustering problems there are multiple data sources which are available. Although each one could individually be used for clustering, exploiting information from all data sources together can be relevant to find a clustering that is more accurate. Here a new model is proposed for clustering when two data sources are available. This model is called Binary View Kernel Spectral Clustering (BVKSC) and is based on a constrained optimization formulation typical to Least Squares Support Vector Machines (LS-SVM). The model includes a coupling term, where the weights of the two different data sources are coupled in the primal model. This coupling term makes it possible to exploit the additional information from each other data source. Experimental comparisons with a number of similar methods show that using two data sources can improve the clustering results and that the proposed method is competitive in performance to other state-of-the-art methods.

1 Introduction

In various application domains data from different sources are available. Many real-world datasets have representations in the form of multiple data sources (also called views) [1]. For example, web pages consist of both the page content (text) and hyperlink information [2], images consist of the pixel arrays but can also have captions associated with them [3], for social networks one could use the user profile but also the friend links [4], and so on. Although each of the data sources by itself might already be sufficient for a given learning task, additional data sources often provide complementary information to each other which can lead to an improved performance [5]. In this paper a new clustering model is introduced, called *Binary View Kernel Spectral Clustering* (BVKSC), that performs clustering when two different data sources are available.

Spectral clustering methods make use of the eigenvectors of some normalized affinity matrix derived from the data to divide a data set into natural groups, such that points within the same group are similar and points in different groups are dissimilar to each other [6, 7, 8]. Kernel Spectral Clustering (KSC) [9] is a well known clustering technique that represents a spectral clustering formulation as a weighted kernel PCA problem, casted in the LS-SVM framework [10].

This paper shows how the clustering performance achieved by KSC on a individual data source can be improved by exploiting information from two different data sources. This is done by integrating two KSC models in the joint BVKSC approach.

2 Binary View Kernel Spectral Clustering

In this section the model *Binary View Kernel Spectral Clustering* (BVKSC) is introduced. This is an extension to KSC where two data sources are used. When training on one data source, the other data source is taken into account by introducing a coupling term in the primal model. By $^{[1]}$ and $^{[2]}$ we denote respectively the first and the second source.

Given training data $\mathcal{D}^{[1]} = \{x_i^{[1]}\}_{i=1}^N \in \mathbb{R}^{d^{[1]}}$ en $\mathcal{D}^{[2]} = \{x_i^{[2]}\}_{i=1}^N \in \mathbb{R}^{d^{[2]}}$ and the number of clusters k, the primal formulation of the BVKSC model is:

$$\begin{aligned}
\min_{\substack{w^{[1]^{(l)}}, w^{[2]^{(l)}}, \\ e^{[1](l)}, e^{[2]^{(l)}}, \\ b_l^{[1]}, b_l^{[2]}}} \; & J = \frac{1}{2}\sum_{l=1}^{k-1} w^{[1]^{(l)T}} w^{[1]^{(l)}} - \frac{1}{2N}\sum_{l=1}^{k-1}\gamma_l e^{[1]^{(l)T}} D^{[1]^{-1}} e^{[1]^{(l)}} \\
& + \frac{1}{2}\sum_{l=1}^{k-1} w^{[2]^{(l)T}} w^{[2]^{(l)}} - \frac{1}{2N}\sum_{l=1}^{k-1}\gamma_l e^{[2]^{(l)T}} D^{[2]^{-1}} e^{[2]^{(l)}} - \rho\sum_{l=1}^{k-1} w^{[1]^{(l)T}} w^{[2]^{(l)}} \\
\text{s.t.} \quad & e^{[1]^{(l)}} = \Phi^{[1]} w^{[1]^{(l)}} + b_l^{[1]} 1_N, \\
& e^{[2]^{(l)}} = \Phi^{[2]} w^{[2]^{(l)}} + b_l^{[2]} 1_N, \qquad l = 1, \ldots, k-1
\end{aligned}$$

where $e^{[1]^{(l)}} \in \mathbb{R}^N$ and $e^{[2]^{(l)}} \in \mathbb{R}^N$ are the projections, $l = 1, \ldots, k-1$ indicate the score variables needed to encode k clusters, $b_l^{[1]}$ and $b_l^{[2]}$ are bias terms, $D^{[1]^{-1}} \in \mathbb{R}^{N\times N}$ and $D^{[2]^{-1}} \in \mathbb{R}^{N\times N}$ are the inverse of the degree matrices $D^{[1]}$ and $D^{[2]}$ with $D_{ii}^{[v]} = \sum_j \varphi^{[v]}(x_i^{[v]})^T \varphi^{[v]}(x_j)$ for $v = 1, 2$ and $\gamma_l \in \mathbb{R}^+$ are regularization constants. $\Phi^{[v]} \in \mathbb{R}^{N\times d_h^{[v]}}$ is a feature matrix with $\Phi^{[v]} = [\varphi^{[v]}(x_1^{[v]})^T; \ldots; \varphi^{[v]}(x_N^{[v]})^T]$ where $\varphi^{[v]} : \mathbb{R}^{d^{[v]}} \to \mathbb{R}^{d_h^{[v]}}$ is the mapping to the high dimensional feature space, for $v = 1, 2$. Here it is imposed that $d_h^{[1]} = d_h^{[2]}$. The primal model consist of taking two times the same formulation as seen in the primal model of KSC but with an added term $-\rho\sum_{l=1}^{k-1} w^{[1]^{(l)T}} w^{[2]^{(l)}}$, namely the *coupling term*. ρ is an additional regularization constant and will be called the *coupling parameter*. The coupling term describes the correlation between the weights of the two sources, which is maximized. In this way the model performs clustering for each data source while exploiting the information from the other source.

By taking the Lagrangian of the primal problem, deriving the KKT optimality conditions and eliminating the primal variables, the dual problem results in the following generalized eigenvalue problem:

$$\left[\begin{array}{c|c} M_{D^{[1]}}\Omega^{[1,1]} & \rho M_{D^{[1]}}\Omega^{[1,2]} \\ \hline \rho M_{D^{[2]}}\Omega^{[2,1]} & M_{D^{[2]}}\Omega^{[2,2]} \end{array}\right] \left[\begin{array}{c} \alpha^{[1]^{(l)}} \\ \hline \alpha^{[2]^{(l)}} \end{array}\right] = \frac{1}{\gamma_l} N(1-\rho^2) \left[\begin{array}{c|c} D^{[1]} & 0 \\ \hline 0 & D^{[2]} \end{array}\right] \left[\begin{array}{c} \alpha^{[1]^{(l)}} \\ \hline \alpha^{[2]^{(l)}} \end{array}\right]$$

where $M_{D^{[v]}} = I_N - \frac{1}{1_N^T D^{[v]^{-1}} 1_N} 1_N 1_N^T D^{[v]^{-1}}$ for $v = 1, 2$ are centering matrices and where $\alpha^{[1]^{(l)}}$ and $\alpha^{[2]^{(l)}}$ are dual variables. The kernel matrices $\Omega^{[1,1]} = \Phi^{[1]}\Phi^{[1]^T}$ and $\Omega^{[2,2]} = \Phi^{[2]}\Phi^{[2]^T}$ capture the similarity between data of the same source. The kernel matrices $\Omega^{[1,2]} = \Phi^{[1]}\Phi^{[2]^T}$ and $\Omega^{[2,1]} = \Phi^{[2]}\Phi^{[1]^T}$ capture the

two different sources. The eigenvalues associated with this eigenvalue problem are $1/\gamma_l$ (as defined by Alzate and Suykens [9]) and ρ is a parameter to be tuned.

Notice that when $\rho = 0$ the dual problem equals two separate KSC problems. This means that when ρ is chosen to be 0, the model equals KSC being applied on both data sources separately.

Since the dimensions of the data sources might be different the kernel trick, as applied by KSC, is not applicable here. A solution is to explicitly define the feature maps $\varphi^{[1]}$ and $\varphi^{[2]}$ as follows:

$$\varphi^{[1]}(x^{[1]}) = \begin{bmatrix} K^{[1]}(x_1^{[1]}, x^{[1]}) \\ \vdots \\ K^{[1]}(x_N^{[1]}, x^{[1]}) \end{bmatrix}, \varphi^{[2]}(x^{[2]}) = \begin{bmatrix} K^{[2]}(x_1^{[2]}, x^{[2]}) \\ \vdots \\ K^{[2]}(x_N^{[2]}, x^{[2]}) \end{bmatrix}.$$

Since $\Omega^{[1,2]} = \Phi^{[1]}{\Phi^{[2]}}^T$, the ij-th element of the matrix $\Omega^{[1,2]}$ is defined as: $\Omega_{ij}^{[1,2]} = \varphi^{[1]}(x_i^{[1]})^T\varphi^{[2]}(x_j^{[2]})$ and similarly for $\Omega_{ij}^{[2,1]} = \varphi^{[2]}(x_i^{[2]})^T\varphi^{[1]}(x_j^{[1]})$, $\Omega_{ij}^{[1,1]} = \varphi^{[1]}(x_i^{[1]})^T\varphi^{[1]}(x_j^{[1]})$ and $\Omega_{ij}^{[2,2]} = \varphi^{[2]}(x_i^{[2]})^T\varphi^{[2]}(x_j^{[2]})$. Since the two 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 defined per data source, it is possible to choose a different kernel function for each source.

The cluster indicators for a certain training sample $\{x_i^{[1]}, x_i^{[2]}\}$ are $\text{sign}({e_i^{[1]}}^{(1)})$, $\ldots$, $\text{sign}({e_i^{[1]}}^{(k-1)})$, $\text{sign}({e_i^{[2]}}^{(1)})$, $\ldots$, $\text{sign}({e_i^{[2]}}^{(k-1)})$. The cluster assignment can be done in two ways:

1. Separately for each data source. Hence two codebooks $\mathcal{C}^{[1]} = \{c_p^{[1]}\}_{p=1}^k$ and $\mathcal{C}^{[2]} = \{c_p^{[2]}\}_{p=1}^k$ are created and the result will be a separate cluster assignment for each data source and these can differ from each other.
2. Together on both data sources. A set of new score variables is defined as $e_{\text{total}}^{(l)} = \beta {e^{[1]}}^{(l)} + (1-\beta){e^{[2]}}^{(l)}$. Only one codebook $\mathcal{C} = \{c_p\}_{p=1}^k$ is created and the cluster assignments for both data sources are performed using these new score variables. The value of β can be 0.5 to take the average, or a value for β can be calculated based on the error covariance matrix (where the error is computed in an unsupervised manner through the silhouette value). Here β is chosen so that it minimizes the error, similarly to how it is done for committee networks [11].

Finally following from the KKT conditions, for out-of-sample test data the projections, and hence the clustering indicators, can be calculated as follows:

$$e_t^{[1]^{(l)}} = \frac{\Omega_{\text{test}}^{[1,1]}\alpha^{[1]^{(l)}} + \rho\Omega_{\text{test}}^{[1,2]}\alpha^{[2]^{(l)}}}{(1-\rho^2)} + b_l^{[1]}1_{N_{\text{test}}}$$
$$e_t^{[2]^{(l)}} = \frac{\Omega_{\text{test}}^{[2,2]}\alpha^{[2]^{(l)}} + \rho\Omega_{\text{test}}^{[2,1]}\alpha^{[1]^{(l)}}}{(1-\rho^2)} + b_l^{[2]}1_{N_{\text{test}}}$$

where $l = 1, \ldots, k-1$. $\Omega_{\text{test}}^{[1,1]} \in \mathbb{R}^{N_{\text{test}} \times N}$, $\Omega_{\text{test}}^{[2,2]} \in \mathbb{R}^{N_{\text{test}} \times N}$, $\Omega_{\text{test}}^{[1,2]} \in \mathbb{R}^{N_{\text{test}} \times N}$ and $\Omega_{\text{test}}^{[2,1]} \in \mathbb{R}^{N_{\text{test}} \times N}$ are the kernel matrices evaluated using the test data

with $\Omega_{\text{test}}^{[1,1]} = \Phi_{\text{test}}^{[1]}\Phi^{[1]^T}$, $\Omega_{\text{test}}^{[2,2]} = \Phi_{\text{test}}^{[2]}\Phi^{[2]^T}$, $\Omega_{\text{test}}^{[1,2]} = \Phi_{\text{test}}^{[1]}\Phi^{[2]^T}$ and $\Omega_{\text{test}}^{[2,1]} = \Phi_{\text{test}}^{[2]}\Phi^{[1]^T}$. The cluster assignment is done in the same manner as for the training phase, so either separately or together (here the same β value is used as in the training phase).

3 Experiments

In this section the results of BVKSC are shown on three datasets and compared with two state-of-the-art methods introduced in [3] namely, Pairwise Co-regularization and Centroid-Based Co-regularization multi-view spectral clustering [1]. In addition, the results are also compared to the best single KSC model, i.e., the best result obtained by performing KSC on the separate data sources. The first dataset used is a synthetic dataset [3] with $d^{[1]} = d^{[2]} = 2$ and $N = 1000$. The second dataset is called the UCI Handwritten digits dataset [12] for which $d^{[1]} = 76$, $d^{[2]} = 216$ and $N = 2000$. The Reuters Multilingual dataset [13] is the third and last dataset, where $d^{[1]} = 21531$, $d^{[2]} = 24892$ and $N = 1200$. More information on these datasets can be found in the work of Kumar et al. [3].

The results obtained by BVKSC depend on the choice of the kernel function and its parameters (as for KSC) and on the choice of the coupling parameter ρ. Since the BVKSC model allows for two different feature maps, two different kernel parameters (one for each source) are tuned.

For the synthetic and the handwritten digits dataset the RBF kernel is chosen. For the Reuters dataset an RBF kernel would not be an appropriate choice because of the high dimensionality of the data. Instead, a normalized polynomial kernel of degree 1 (linear) and 2 were considered. The tuning of the kernel parameters and the coupling parameter ρ is done by means of 2-fold cross validation where simulated annealing is used to optimize the performance. This is repeated five times, so for a certain set of parameters there are ten different clusterings, depending on the chosen training/validation set. The mean of the performance criterion over these ten runs is then used as the evaluation metric for a certain set of parameters. To assess the performance of the model the BLF [9] and BAF [14] criteria were considered [2]. The criterion is evaluated for each data source and the total performance of the model is the mean of both.

To evaluate the cluster quality NMI [15] is used. The criterion is evaluated for each data source and the total performance is the mean of the two values.

The results are depicted in Table 1. The table shows the results found for two versions of BVKSC, one where the cluster assignment is done separately for each source (denoted by $(e^{[1]}, e^{[2]})$) and one where the cluster assignment is done together for both sources (denoted by (e_{total})). It also shows the results with the Co-regularized methods from [3] where the kernel and co-regularization parameters are tuned in the same manner as described before for BVKSC. The table also depicts the runtime corresponding to clustering on the entire dataset.

[1] We thank the authors for providing the code to do the experiments.

[2] Only the best obtained results (obtained either with BLF or BAF) will be reported.

Table 1: NMI and runtime (in seconds) results on test data. The highest NMI values, and hence the best performing methods, are indicated in bold.

Method	Synth data		Handwritten		Reuters	
	NMI	Time (s)	NMI	Time (s)	NMI	Time (s)
Best Single KSC	0.232	0.416	0.559	0.755	0.325	4.79
Co-regularized (P)[3]	0.299	86.6	**0.673**	61.4	0.375	51.9
Co-regularized (C)[3]	0.338	58.6	0.659	66.9	0.360	51.6
BVKSC ($e^{[1]}, e^{[2]}$)	0.324	5.38	0.559	7.07	**0.379**	41.1
BVKSC (e_{total})	**0.350**	2.30	0.583	22.5	0.342	1.23e+03

For all three datasets, the proposed model BVKSC performs better than (or at least equally good as) the best single KSC model. This indicates that the additional information from a second source improves the result. For the synthetic dataset as well as for the Reuters Multilingual dataset, BVKSC is also able to perform better than the methods from [3].

For the UCI Handwritten digits dataset the first version of BVKSC (cluster assignment is done separately for each source) performs equally well as the best single KSC model. This result was obtained with $\rho = 0$ and is thus the special case of BVKSC where the model equals KSC being applied on both sources separately. This indicates that this version of BVKSC is not able to improve the result. This can be explained by looking into the performance of KSC on both sources separately. While the best performance of KSC on the first source results in a NMI value of 0.559, for the best performance on the second source the NMI equals 0.0146. This is a very low cluster quality and it seems that the second data source does not contain much useful information. However, even for this dataset, the second version of BVKSC (cluster assignment is done together for both sources) is still able to improve the results.

Table 1 also shows that the runtime increases when using an additional source, but also that BVKSC is faster than the other multi-source algorithms. The only time BVKSC is slower, is for the second version on the Reuters dataset. The high runtime here is due to the high dimensionality of the Reuters data and the use of silhouette to determine β. In fact, the time to compute the silhouette values is about 92% of the total runtime. If computational cost is critical, this can be circumvented by using another criteria, using a fixed β or using the first version of BVKSC.

4 Conclusion and perspectives

In this paper the problem of exploiting information from two data sources when performing clustering is addressed. Although each of the data sources by itself might already be sufficient for clustering, the aim is to improve the performance by incorporating information from other sources. In this paper the model Binary View Kernel Spectral Clustering is introduced, which combines information from two data sources when performing clustering. The model is casted in the LS-

SVM framework [10] where weight coupling is done in the primal model by means of a coupling term. The model is tested on three datasets. The obtained results show the improvement of using multiple data sources. Most of the results are also better than the results obtained for other methods that perform clustering on multiple data sources.

In future work we aim to extend the model in order to exploit information from more than two data sources. For this we may need to look into other possible coupling schemes.

Acknowledgments

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC AdG ADATADRIVE- B (290923). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information. Research Council KUL: CoE PFV/10/002 (OPTEC), BIL12/11T; PhD/Postdoc grants Flemish Government: FWO: projects: G.0377.12 (Structured systems), G.088114N (Tensor based data similarity); PhD/Postdoc grant iMinds Medical Information Technologies SBO 2015 IWT: POM II SBO 100031 Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO, Dynamical systems, control and optimization, 2012-2017).

References

[1] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan. Multi-view clustering via canonical correlation analysis. *ICML*, pages 129–136, 2009.

[2] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. *COLT*, pages 92–100, 1998.

[3] A. Kumar, P. Rai, and H. Daume. Co-regularized multi-view spectral clustering. *NIPS*, pages 1413–1421, 2011.

[4] Y. Yang, C. Lan, X. Li, J. Huan, and B. Luo. Automatic social circle detection using multi-view clustering. *CIKM*, pages 1019–1028, 2014.

[5] J. Du, C. X. Ling, and Z-H. Zhou. When does co-training work in real data? *IEEE Transactions on Knowledge and Data Engineering*, 23:788–799, 2010.

[6] A. Y Ng, A. Y Jordan, and Y Weiss. On spectral clustering: Analysis and an algorithm. *NIPS*, 2:849–856, 2002.

[7] U von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[8] F R. Chung. *Spectral Graph Theory*, volume 92. Providence, RI, USA: AMS, 1997.

[9] C. Alzate and J. A. K. Suykens. Multiway spectral clustering with out-of-sample extensions through weighted kernel PCA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):335–347, 2010.

[10] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[11] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[12] M. Lichman. UCI machine learning repository, 2013.

[13] M-R. Amini, N. Usunier, and C. Goutte. Learning from multiple partially observed views - an application to multilingual text categorization. *NIPS*, pages 28–36, 2009.

[14] R. Mall, R. Langone, and J. A. K. Suykens. Kernel spectral clustering for big data networks. *Entropy*, 15:1567–1586, 2013.

[15] A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.