# Adaptive multi-view clustering via cross trace lasso

Dong Wang, Ran He, Liang Wang and Tieniu Tan
Institute of Automation, Chinese Academy of Sciences
{dwang, rhe, wangliang, tnt}@nlpr.ia.ac.cn

## Abstract

*We propose a novel multi-view clustering method by learning auto-regression problems under structural constraints and treating the regression coefficients as new feature representations for the cluster partition. In particular, we take the data intrinsic correlation structure into account. Correlated data under one view tend to be also related under another view and are likely to fall into the same group. Therefore we pair the data matrix from one view and the regression coefficient from a different view together to meet a trace Lasso constraint, which adaptively adjusts the sparsity of regression coefficients in order to promote consistent data correlations across views. Then a joint low-rank constraint is further imposed to encourage similar regression coefficients for the same samples under distinct views. Finally, we develop an effective algorithm to optimize the objective function. And experimental results demonstrate that our method is useful and fairly competitive compared with other state-of-the-art multi-view clustering methods.*

## 1. Introduction

Multi-view clustering, whose goal is to discover the true underlying cluster membership of the data points given their feature representations under multiple views, has been extensively studied over the past few years. A view is often loosely referred to as one kind of feature modality, which provides a view-specific description of the data. Compared with single-view clustering, multi-view clustering methods try to benefit from the multiple sources of information available, in hope of finding a more reliable and consistent cluster partition [20]. As the volume of multimedia data accumulated on the web becomes bigger and bigger, we can easily get access to data that have multiple views. For example, user-generated contents on the web usually involve textual, visual or audio information at the same time. Information from each view or modality serves as an additional cue that reflects the characteristic of the same content.

Over the past few years a series of methods have been proposed to obtain more precise clustering results by taking advantage of the potential complementary information across multiple views or modalities. Concretely, one category of the state-of-the-art methods is to directly unify the multi-view information in the clustering process. For instance, in [15] a co-training flavored spectral clustering algorithm was proposed to encourage the clustering agreement between views. Another one is [16] which attempted to regularize on the eigenvectors of view-specific graph laplacians and achieve consistent clusters across views. Another line of research is to first learn a latent representation for multi-view data and then perform clustering on such representation to learn the partition. A notable one is [18] which employed matrix factorization to discover a common latent structure shared by all views and give rise to compatible clustering results. Besides, CCA based multi-view clustering methods also fall into this category [8] [5]. Yet another line of research is by fusing the clustering results obtained from individual views toward a consensus [7] [13].

In this paper, inspired by the previous success of subspace clustering like Sparse Subspace Representation (SSC) [11] and Low-Rank Representation (LRR) [17], we adopt the strategy of letting data points linearly represent each other. The advantage of such *auto-regression* method is that we can capture the data affinity by learning a *global distributed* feature representation (representational responses over all the data points), while in the previous methods the data affinity are captured through computing *local pairwise* Euclidean distance in the similarity graph construction.

Moreover, the auto-regression strategy is also an effective way of capturing the group structure. Usually the regression coefficient is encouraged to be sparse in a sense that only data points of the same cluster should be used to represent each other. Therefore the degree of such sparsity decides which candidate points are selected to form a group and thus reflects the underlying cluster structure. But when data are highly correlated, if the sparse constraint is too strong, perhaps only one of correlated data points is selected for the reconstruction. But our goal is to assign all the correlated data points to the same cluster. Intuitively, correlated data under one view tend to be also correlated under another view and are likely to fall into the same group.

Therefore we propose to pair the data matrix from one view and the regression coefficient from a different view together to meet a trace lasso constraint, which adaptively adjusts the sparsity of regression coefficients in order to promote consistent data correlations across views. For the same set of samples, their regression coefficients under different views are further enforced to follow a joint low-rank constraint, anticipating that the same data should have similar distributed feature representations under any view. This will lead to a more stable and consistent clustering result.

The contributions of this paper are summarized as follows:

• We propose a novel multi-view clustering method which exploits the overall data correlation across views. The resulting sparsity-adaptive regression coefficients can better capture the underlying intra & inter cluster affinities and thus improve the clustering performance.

• We develop an effective algorithm based on iterative least squares to optimize the objective function. Experimental results on common benchmark databases validate the usefulness of the proposed multi-view clustering method.

## 2. Proposed method

In this section, we first introduce some preliminaries about trace Lasso to better understand our model. Then the formulation and optimization of the proposed method will be illustrated in great details.

### 2.1. Preliminaries for trace Lasso

Trace Lasso is developed to adapt to the intrinsic data correlation [12]. It performs in between $l_1$ norm or $l_2$ norm, depending on how correlated the original data are. It is defined as follows:

$$\Omega(w) = ||X\text{Diag}(w)||_* \tag{1}$$

Specifically, when the data matrix is column-wise normalized to have unit norm, $X\text{Diag}(w)$ can be further expanded as follows:

$$X\text{Diag}(w)|| = \sum_{i=1}^{n} |w_i| \, (\text{sign}(w_i)x_i)e_i^T \tag{2}$$

Consider the extreme case where the data points are all uncorrelated and orthogonal to each other (i.e. $(X)^T X = I$ ). The above equation gives the singular value decomposition of $X\text{Diag}(w)$. Then, trace Lasso is equal to $l_1$ norm:

$$||X\text{Diag}(w)||_* = ||\text{Diag}(w)||_* = \sum_{i=1}^{n} |w_i| = ||w||_1 \tag{3}$$

On the contrary, when the data are highly correlated and suppose they are the same (i.e. $X = x_1\mathbf{1}^T$, $(X)^T X = \mathbf{1}\mathbf{1}^T$), trace Lasso is then reduced to $l_2$ norm.

$$||X\text{Diag}(w)||_* = ||x_1 w^T||_* = ||x_1||_2||w||_2 = ||w||_2 \tag{4}$$

Under other circumstances, trace Lasso behaves in between $l_1$ norm and $l_2$ norm.

$$||w||_2 \leqslant \; ||X\text{Diag}(w)||_* \leqslant \; ||w||_1 \tag{5}$$

By coupling the data matrix and the coefficient together, trace Lasso becomes adaptive to the data correlation. One nice property about trace Lasso is that it interpolates between $l_1$ and $l_2$ regularizers, which hardly makes the resulting coefficient either too sparse or too dense. Thus correlated data are likely to be grouped together.

### 2.2. Formulation of the proposed method

In our model, under each view similar data points are used to linearly reconstruct each other. Given the data matrix $X^I \in \mathbb{R}^{d_I \times n}$ and regression coefficient matrix $Z^I \in \mathbb{R}^{n \times n}$ where $I$ denotes one of the two views $A$ and $B$, $d_I$ is the feature dimension of view $I$ and $n$ is the number of data points, $Z_i^A$ (or $Z_i^B$ ) is the $i$-th column of matrix $Z^A$ (or $Z^B$). Then our proposed method is formulated as follows:

$$\begin{aligned}\min_{Z^A, Z^B} \sum_{I \in \{A,B\}} ||X^I - X^I Z^I||_F^2 + \beta \sum_{i=1}^{n} (||X^B Diag(Z_i^A)||_* \\ + ||X^A Diag(Z_i^B)||_*) + \gamma ||(Z^A, Z^B)||_*\end{aligned} \tag{6}$$

For the ease of exposition, here we give the formulation of our method in a two-view scenario. However, our method can be easily generalized to the scenarios involving more than three views. The optimization procedure can be derived by following the same alternative iteration strategy.

In our formulation, unlike CCA-based methods trying to map data $X$ onto the latent subspace or MF based methods trying to decompose $X$ and obtain latent features, we aim to learn new feature representations $Z$ that can not only reconstruct the data itself, but also reflect the underlying data correlation across views. If we think of all the data points as a codebook, then the resulting $Z$ is the global distributed feature responses on this codebook. For any regression target, candidate points which are similar to the target tend to have larger response entries in $Z$ and be selected to play a more significant role in the regression.

In order to obtain a consistent cluster partition, correlated data points under one view are also expected to stay correlated under other views. Therefore in our formulation we use trace Lasso to enforce that the distributed feature representation derived from view $A$ can adapt to the data correlation under view $B$, and vice versa. Such cross regularization makes the new feature representation $Z_A$ (or $Z_B$) adapt to the data correlation of both its own view and other views. So the complementary information among views

is exchanged on the new feature representation level. Finally in the last term of our formulation, $Z_A$ and $Z_B$ are concatenated to form a larger matrix and are further subject to a joint low-rank constraint, which means corresponding columns of $Z_A$ and $Z_B$ are asked to be similar. The intuition is that no matter under which view, the same group of candidates should be selected to play equally important roles in reconstructing the target.

$\beta$ and $\gamma$ are the hyper-parameters that control the trade-off between corresponding terms. Once $Z_A$ and $Z_B$ are obtained, we average them by letting $Z = (|Z_A| + |Z_B|)/2$. Then a spectral clustering algorithm like [19] is applied on $Z$ to complete the final clustering procedure.

### 2.3. Solution to the proposed method

In terms of solving the proposed objective, it is not easy to optimize (6) directly given the existence of a trace norm regularizer. So we rewrite our objective by following a well established variational formulation for trace norm [6] [14], in which the statement below holds true for a matrix $M$:

$$||M||_* = \frac{1}{2} \inf_{S \succcurlyeq 0} \; tr(M^T S^{-1} M) + tr(S) \tag{7}$$

where the infimum is obtained for $S = (MM^T)^{1/2}$.

In the outer loop of our algorithm, we alternatively solve for one of the representational matrix $Z^A$ or $Z^B$ while keeping the other one fixed. In light of the results from (7), when we optimize the objective with respect to $Z^A$ in a column-wise fashion, (6) can be simplified into the following

$$\min_{Z^A} \sum_{i=1}^{n} ||X_i^A - X^A Z_i^A||_2^2 + \beta ||X^B Diag(Z_i^A)||_* + \gamma ||(Z^A, Z^B)||_* \tag{8}$$

Suppose $\Omega_1 = ||X^B Diag(Z_i^A)||_*$ and apply (7) to (8), we have the following:

$$\Omega_1 = \frac{1}{2} \inf_{S_1 \succcurlyeq 0} \; (Z_i^A)^T Diag(diag[(X^B)^T S_1^{-1} X^B]) Z_i^A + tr(S_1) \tag{9}$$

Here $S_1 = (X^B [Diag(Z_i^A)]^2 (X^B)^T + \mu_1 I)^{1/2}$ can be seen as an intermediate variable during the optimization.

Suppose $\Omega_2 = ||X^A Diag(Z_i^B)||_*$ and apply (7) to (6) which this time minimizes with respect to $Z^B$. We have

$$\Omega_2 = \frac{1}{2} \inf_{S_2 \succcurlyeq 0} \; (Z_i^B)^T Diag(diag[(X^A)^T S_2^{-1} X^A]) Z_i^B + tr(S_2) \tag{10}$$

Similarly, $S_2 = (X^A [Diag(Z_i^B)]^2 (X^A)^T + \mu_2 I)^{1/2}$ is another intermediate variable during the optimization.

To simplify the joint low-rank constraint, we rewrite this term based on (7) and obtain:

$$||(Z^A, Z^B)||_* = \frac{1}{2} \inf_{S_3 \succcurlyeq 0} \; tr\left( \begin{bmatrix} (Z^A)^T \\ (Z^B)^T \end{bmatrix} S_3^{-1} [Z^A, Z^B] \right) + tr(S_3) \tag{11}$$

After some further expansion and simplification, the above equation boils down to:

$$||(Z^A, Z^B)||_* = \frac{1}{2} \inf_{S_3 \succcurlyeq 0} \; \sum_{i=1}^{n} [(Z_i^A)^T S_3^{-1} Z_i^A + (Z_i^B)^T S_3^{-1} Z_i^B] + tr(S_3) \tag{12}$$

Here $S_3 = [Z^A (Z^A)^T + Z^B (Z^B)^T + \mu_3 I]^{1/2}$ is another auxiliary variable during the alternative optimization process:

For the ease of illustration, we define $D^B = Diag(diag[(X^B)^T S_1^{-1} X^B])$. Finally, putting all the pieces together, when we optimize (6) with respect to the $i$-th column of $Z^A$, we get the following re-weighted least squares problem:

$$\min_{Z_i^A} \sum_{i=1}^{n} ||X_i^A - X^A Z_i^A||_2^2 + \beta (Z_i^A)^T D^B Z_i^A + \gamma (Z_i^A)^T S_3^{-1} Z_i^A \tag{13}$$

It is not difficult to derive the solution to the above problem:

$$Z_i^A = [(X^A)^T X^A + \beta D^B + \gamma S_3^{-1}]^{-1} (X^A)^T X_i^A \tag{14}$$

Lisewise, when optimizing the objective with respect to $Z_i^B$, let $D^A = Diag(diag[(X^A)^T S_2^{-1} X^A])$ and we have the following:

$$Z_i^B = [(X^B)^T X^B + \beta D^A + \gamma S_3^{-1}]^{-1} (X^B)^T X_i^B \tag{15}$$

## 3. Experimental results

In this section, we evaluate our method on widely used benchmark databases and compare with a series of baselines in order to validate the usefulness of the proposed model.

### 3.1. Databases

**Movies617 dataset** [3] consists of 617 movies with 17 labels extracted from IMDb. The two views are the 1878 keywords and the 1398 actors with a keyword used for at least 2 movies and an actor appeared in at least 3 movies.

**Animal dataset** [2] consists of 30475 images of 50 animals with six pre-extracted features for each image. Three kinds of features, namely PyramidHOG (PHOG), color-SIFT and SURF, are chosen as three views. We select the first ten categories with each including randomly chosen 50 samples as a subset for evaluation.

**UCI Handwritten Digit dataset** [1] consists of features of handwritten digits (0–9). The dataset is represented in

terms of six features and contains 2000 samples with 200 in each category. Similar to [16], we select the 76 Fourier coefficients of the character shapes and the 216 profile correlations as two views of the original dataset.

**Pascal VOC 2007 dataset** [4] consists of 20 categories with a total of 9,963 images. We use the Color feature and Bow feature as two-view visual representation. Furthermore, those images with multiple categories are removed, thus leaving 5,649 images for evaluation.

**NUS WIDE dataset** [10] consists of 269,648 images of 81 categories collected from Flickr. We select 500 images from each of the five classes with the most number of images for evaluation. Six types of low level features are given and we use color correlogram and wavelet texture as two-view representations for multi-view clustering.

### 3.2. Experimental settings

We extensively compare our method with many representative baselines including 1) *S_Spectral*: Use spectral clustering in [19] to cluster each view's data and select the best clustering result. 2) *S_LowRank*: Use only single-view low-rank representation to construct the affinity matrix and then apply spectral clustering in [19] to cluster the dataset. We also report the best clustering results. 3) *Combined*: Concatenate features from two views and apply low-rank representation on the combined feature to perform clustering. 4) *PairwiseSC*: [16] co-regularize the eigenvectors of all views' Laplacian matrices. 5) *Co_Training*: [15] Alternately modify one view's graph structure using the other view's information. 6) *Multi_NMF*: [18] A multi-view non-negative matrix factorization method to group the multi-view data. Note that this method is not applicable on NUS dataset since it requires all non-negative input features. 7) *Multi_SS*: [21] A structure sparsity based multi-view clustering and feature learning framework. The parameters in these methods are carefully selected in order to achieve their best results. Once K-means is used, it is run 20 times with random initialization. To measure the clustering results, we use accuracy (ACC) and normalized mutual information (NMI). Readers can refer to [9] for more details about such measures. Both mean and standard deviation are reported.

### 3.3. Experimental results and analysis

The experimental results in terms of both NMI and accuracy are listed in Table 1 and 2 respectively. In terms of NMI, our method performs better than all the other compared baselines. First of all, simple baselines $S\_Spectral$ and $S\_LowRank$ fail to give good results as they only rely on one source of information for the data partition. Then the third baseline, which applies low-rank representation on a simple concatenation of features from multiple views, tends to improve clustering performance to some extent, but is still less competitive than our method. One possible reason is that feature vectors from different views may appear to be heterogenous, which could make correlated data under different views less correlated after feature concatenation. While our method solves auto-regression problems using homogeneous feature vectors under each view and cross-regularizes the regression coefficients to adapt to the intrinsic data correlation across views. Such data correlation information is circulated among different views through the resulting regression coefficients, which makes better use of the data affinity and finds precise clusters. $Multi\_SS$ solves the problem of integrating heterogeneous feature sets by learning feature weights via structured sparsity norms. It encourages sparsity between views, but is less adaptive to data correlation within each view and thus less flexible than our method. For $PairwiseSC$, $Co\_Training$ and $Multi\_NMF$, they capture the data affinity by either constructing similarity graphs or learning latent representations via matrix decomposition. They are different from our method which treats all data points as a global basis and learns distributed feature representations under the auto-regression framework. Our method beats them possibly because the regression coefficients in our formulation reflect the intra & inter cluster structures more precisely via the adaptive ability of trace Lasso.

## 4. Conclusion

In the multi-view setting, data can have various feature modalities at the same time, which may not guarantee correlated data under one view still stay correlated under a different view. To mitigate such uncertainty, we have proposed a trace Lasso regularized framework that adapts to the data correlation from all views. Our method flexibly adjusts the sparsity of the regression coefficients and makes sure that correlated data should fall into the same cluster. The resulting regression coefficient serves as a new distributed feature representation over the basis of all data points. An additional joint low-rank constraint is imposed to let the same samples have similar distributed feature representations across views. Experimental results on a series of datasets have demonstrated the usefulness of our proposed method.

## References


[1] http://archive.ics.uci.edu/ml/datasets/Multiple+Features. 3

[2] http://attributes.kyb.tuebingen.mpg.de/. 3

[3] http://membres-lig.imag.fr/grimal/data.html. 3

[4] http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/. 4

[5] U. Ahsan and I. Essa. Clustering social event images using kernel canonical correlation analysis. In *CVPR Workshops*, pages 814–819, 2014. 1

| ACC(%) | Digits | Movies617 | VOC | Animal | NUS |
|---|---|---|---|---|---|
| S_Spectral | 66.37(4.44) | 25.70(1.13) | 15.64(0.43) | 27.21(1.50) | 33.85(0.18) |
| S_LowRank | 66.53(5.31) | 30.02(1.05) | 17.09(0.41) | 31.71(1.88) | 34.74(0.02) |
| Combined | 71.83(6.19) | 32.93(1.53) | 13.98(0.37) | 32.51(1.00) | 30.28(0.14) |
| PairwiseSC | 80.82(6.30) | 27.89(1.64) | 11.93(0.14) | 31.65(1.59) | 33.07(0.14) |
| Co_Training | 80.22(6.84) | 30.74(1.28) | 14.84(0.33) | 30.35(1.48) | 35.15(1.03) |
| Multi_NMF | 69.24(6.28) | 26.99(1.19) | 12.57(0.23) | 30.56(1.02) | N/A |
| Multi_SS | 72.45(4.10) | 29.60(1.10) | 12.47(0.26) | 32.11(1.86) | 32.39(0.05) |
| **AMCTL** | **82.34(5.89)** | **34.03(1.45)** | **18.57(0.52)** | **34.46(1.69)** | **36.09(1.12)** |

Table 1: Clustering results in terms of accuracy on five benchmark databases.

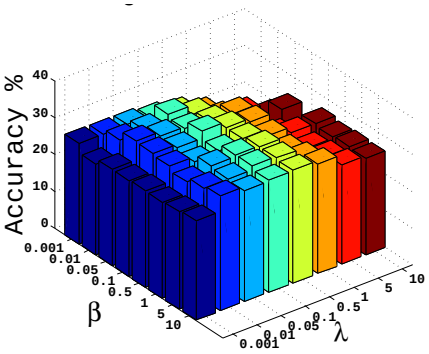


Figure 1: Accuracy vs. parameters $\lambda$ and $\beta$ on the Movies617 database

| NMI(%) | Digits | Movies617 | VOC | Animal | NUS |
|---|---|---|---|---|---|
| S_Spectral | 62.30(1.85) | 25.47(0.85) | 9.34(0.19) | 15.70(0.65) | 6.64(0.15) |
| S_LowRank | 69.79(1.81) | 30.15(0.80) | 7.05(0.18) | 18.42(1.10) | 7.68(0.01) |
| Combined | 70.92(2.03) | 32.11(0.89) | 9.33(0.30) | 20.09(0.58) | 5.14(0.27) |
| PairwiseSC | 75.84(2.37) | 28.04(0.73) | 6.07(0.12) | 19.90(1.51) | 6.87(0.01) |
| Co_Training | 75.90(2.27) | 30.74(1.28) | 9.88(0.21) | 18.98(0.73) | 8.51(0.16) |
| Multi_NMF | 65.05(2.30) | 27.45(0.55) | 6.40(0.19) | 18.77(0.71) | N/A |
| Multi_SS | 74.55(2.49) | 30.09(1.32) | 6.76(0.11) | 21.25(1.76) | 5.63(0.02) |
| **AMCTL** | **79.18(2.71)** | **33.21(1.01)** | **9.90(0.25)** | **23.02(0.92)** | **8.70(0.18)** |

Table 2: Clustering results in terms of NMI on five benchmark databases.

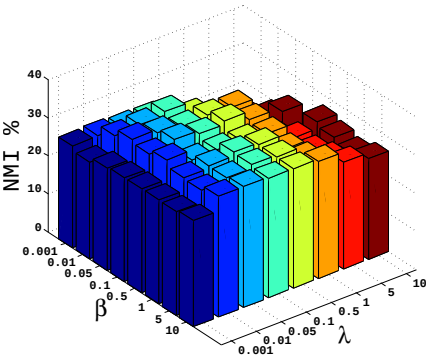


Figure 2: NMI vs. parameters $\lambda$ and $\beta$ on the Movies617 database


[6] A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. In *NIPS*, 2007. 3

[7] E. Bruno and S. Marchand-Maillet. Multiview clustering: A late fusion approach using latent models. *In ACM CRDIR*, pages 736–737, 2009. 1

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

[9] W.-Y. Chen, Y. Song, H. Bai, C.-J. Lin, and E. Y. Chang. Parallel spectral clustering in distributed systems. *IEEE TPAMI*, 33(3):568–586, 2011. 4

[10] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng. Nus-wide: A real-world web image database from national university of singapore. In *ACM CIVR*, 2009. 4

[11] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE TPAMI*, 35(1):2765–2781, 2013. 1

[12] E. Grave, G. R. Obozinski, and F. R. Bach. Trace lasso: a trace norm regularization for correlated designs. In *NIPS*, pages 2187–2195, 2011. 2

[13] D. Greene and P. Cunningham. A matrix factorization approach for integrating multiple data views. *In ECML-KDD*, pages 423–438, 2009. 1

[14] Y. Guo. Convex subspace representation learning from multi-view data. In *AAAI*, 2013. 3

[15] A. Kumar and H. D. Iii. A co-training approach for multi-view spectral clustering. *ICML*, pages 393–400, 2011. 1, 4

[16] A. Kumar, P. Rai, and H. D. Iii. Co-regularized multiview spectral clustering. *NIPS*, pages 1413–1421, 2011. 1, 4

[17] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. *In ICML*, pages 663–670, 2010. 1

[18] J. Liu, C. Wang, J. Gao, and J. Han. Multi-view clustering via joint nonnegative matrix factorization. *SDM*, pages 252–260, 2013. 1, 4

[19] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE TPAMI*, 22(8):888–905, 2000. 3, 4

[20] S. Sun. A survey of multi-view machine learning. *Neural Computing and Applications*, 23(7-8):2031–2038, 2013. 1

[21] H. Wang, F. Nie, and H. Huang. Multi-view clustering and feature learning via structured sparsity. *ICML*, pages 352–360, 2013. 4