Multi-view Spectral Clustering on Conflicting Views

Xiao He[1,2], Limin Li[3], Damian Roqueiro[1,2], and Karsten Borgwardt[1,2]

[1] Department of Biosystems Science and Engineering, ETH Zurich, Switzerland
[2] Swiss Institute of Bioinformatics, Switzerland
{xiao.he,damian.roqueiro,karsten.borgwardt}@bsse.ethz.ch
[3] School of Mathematics and Statistics, Xi'an Jiaotong University, China
liminli@mail.xjtu.edu.cn

Abstract. In a growing number of application domains, multiple feature representations or *views* are available to describe objects. *Multi-view clustering* tries to find similar groups of objects across these views. This task is complicated when the corresponding clusterings in each view show poor agreement (*conflicting* views). In such cases, traditional multi-view clustering methods will not benefit from using multi-view data. Here, we propose to overcome this problem by combining the ideas of multi-view spectral clustering with alternative clustering through kernel-based dimensionality reduction. Our method automatically determines feature transformations in each view that lead to an optimal clustering w.r.t to a new proposed objective function for conflicting views. In our experiments, our approach outperforms state-of-the-art multi-view clustering methods by more accurately detecting the ground truth clustering supported by all views.

Keywords: Multi-view clustering, Alternative clustering, Conflicting views, Kernel dimensionality reduction

1 Introduction

In many application domains, it is commonplace that a single object can be described by multiple feature representations or *views*. We will expect to obtain a clustering of better quality if information on all views is taken into account. *Multi-view clustering* tries to find similar groups of objects across different views, with a number of methods having been proposed in the literature, including Multi-view EM [1], Canonical Correlation Analysis for multi-view clustering [2], Multi-view spectral clustering [3,4,5], Multi-view clustering with unsupervised feature selection [6,7] and Nonnegative Matrix Factorization [8]. All these methods share the assumption of a common clustering structure across views, which is interpreted as having the corresponding clusterings in each view in agreement with a ground truth partitioning.

However, in real-world datasets, certain views may contain subsets of features with varying degrees of relatedness which may lead to multiple non-redundant alternative clustering solutions in each view. For example, while clustering university webpages by text features (words), some words such as 'major', 'position' or 'homework' will lead to a partitioning of webpages into categories such as 'student', 'faculty' and 'course'. Alternatively, other words (e.g. 'biology', 'cell', 'computer science', 'code' etc.) will

lead to a partitioning of webpages by their department of affiliation, which is independent of the categories described before. The final clustering will be closer to one of the partitionings but contaminated by the other one.

Several methods [9,10,11] were proposed to tackle this alternative clustering problem, but they only focus on single-view data. The problem exists in the multi-view setting as well but it has received little attention. When the corresponding clusterings in each view show poor agreement, we say that we have *conflicting* views. The users may only be in favor of the underlying clustering that is closer to the partitioning on one view. In such cases, traditional multi-view clustering methods [1,2,3,5,6,8] will fail to be beneficial or may even be harmful when using multi-view data.

Going back to our example of university webpages, if we consider webpages comprised of two views: a) text (words) and b) hyperlinks, a clustering on the text view will cluster webpages into a partitioning by categories ('student', 'faculty', 'course', etc) since more word features are related to this partition. Suppose users are interested in finding this partition. However, a clustering on the hyperlink view will mainly partition webpages by their department of affiliation. This is due to the fact that, for example, students' webpages may link to the courses for which they are registered while webpages of faculty members are linked to the pages of the courses which they teach. Therefore, the two views conflict and their corresponding partitions are likely to disagree. In addition, as mentioned before, the clustering on the text view is contaminated by word features that lead to a partitioning of webpages by their department of affiliation (partitioned on the hyperlinks view). We consider such an underlying structure (e.g. department) which unduly influences the partitioning to be a *confounder*. There might be useful information in the hyperlinks view, but this is masked by the confounder. To unveil the desired clustering structure across conflicting views, we need to find agreement between patterns across views and correct for the confounder.

Our goal in this article is, therefore, to perform multi-view clustering on conflicting views and to correct for possible confounders. We define a novel objective function that combines ideas of multi-view spectral clustering and alternative clustering and propose a new algorithm `MvKDR` to solve it. More specifically, we project each view onto two different subspaces where two alternative clusterings can be found based on kernel dimensionality reduction [12,13]. With the prior knowledge of which view is more informative, we then try to find a consensus partition by maximizing the agreement between clusterings on one subspace from each view and minimize the agreement between this consensus partition with the alternative clustering on the other subspaces from all views. The motivation behind our method `MvKDR` is that conflicting views lead to disagreement (statistical independence) between the clusterings across different views. We aim to maximize agreement (statistical dependence) between clusterings across all views on the reduced subspaces and correct for possible confounders through the process of finding an alternative clustering in each view.

Our motivating example for solving this problem of conflicting views comes from cancer genomics. Here, patients are described by different views which consist of molecular tests performed on their tissues. These are the expression level of a) genes and b) DNA methylation. The aim is to discover cancer subtypes by clustering the patients. DNA methylation is known to be a mechanism that the cell uses to control gene expres-

sion, and so it is reasonable to expect an intrinsic disease-related clustering across both views. However, we found in our experiments that independent clusterings on the gene expression view and the DNA methylation view show little agreement. This may be due to the fact that not all genes or DNA methylation mechanisms are disease-related. Some biological processes that affect both gene expression and DNA methylation may act as confounders. In our experiments, we perform a survival analysis on gene and DNA methylation expression data of cancer patients to show the effectiveness of our proposed method `MvKDR`.

In regards to related work, and as mentioned before, most of the multi-view clustering techniques in the literature [1,2,3,4,5,6,7,8] do not consider confounding factors and conflicting views, which are the focus of this paper. Christoudias et al. [14] studied the multi-view problem in the presence of view disagreement but in a supervised manner. The most related work are `Pareto` [15] and `MVUFS` [16]. `Pareto` is a multi-objective spectral clustering method based on pareto optimization. However, `Pareto` performs on the Laplacian matrix in the full feature space, which may fail to detect clustering structure that can only be found in subspaces. `MVUFS` tries to do feature selection on the second view when it conflicts with the first one. The proposed `MvKDR` differs from `MVUFS` by considering the correction of confounding factors across views. In our experiments, we compare to `Pareto` and `MVUFS`, as well as other state-of-the-art multi-view clustering approaches on both synthetic and real-world data. Our goal is to show the advantages offered by our proposed method `MvKDR` by more accurately detecting the underlying ground truth clusterings.

The remainder of this paper is organized as follows: in the following section we describe the proposed multi-view spectral clustering model and describe the algorithm in detail. Section 3 contains an extensive experimental evaluation. Section 4 concludes the paper with a summary of our work and future direction.

2 Multi-view spectral clustering on conflicting views

In this section, we first review the co-regularized multi-view spectral clustering framework. We then extend it to our new model `MvKDR` with confounder correction by applying the technique of kernel dimensionality reduction. We finally provide the optimization algorithm for solving the model.

2.1 Co-regularized spectral clustering

Suppose we are given an m-view dataset of $\{X_1, \cdots, X_m\}$, where $X_v \in \mathbb{R}^{n \times d_v}$, n and d_v are the number of samples and features in view v. Suppose the number of true clusters k is known.

The kernel matrix of X_v is denoted as K_v and D_v is the degree matrix of K_v. The normalized graph Laplacian for view v can be written as $L_v = D_v^{-\frac{1}{2}} K_v D_v^{-\frac{1}{2}}$. Based

on this definition, co-regularized spectral clustering CoReg [3] was proposed as:

$$\max_{\text{All } U_v} \sum_{v=1}^{m} tr(U_v^T L_v U_v) + \lambda \sum_{v \neq w} R(U_v, U_w), \quad s.t.\ U_v^T U_v = I \tag{1}$$

where $U_v \in \mathbb{R}^{n \times k}$ with $v \in 1, ..., m$, $R(U_v, U_w) = tr(U_v U_v^T U_w U_w^T)$ is a regularizer that measures the agreement between the embeddings U_v and U_w and λ trades-off the spectral clustering objective and the embedding agreement. The problem can be solved by alternating maximization cycling over the views with all but one U_v fixed. Since alternating maximization converges to a local maximum, CoReg usually starts with an informative view and performs k-means on the final embedding of that view, with the assumption that we have prior knowledge about views.

2.2 Multi-view spectral clustering with kernel dimensionality reduction

In the scenario of conflicting views, the agreement of embeddings from Laplacians obtained in the full space will not be useful, or even worse, it can be harmful. As mentioned before, our idea is to first project each view to a low-dimensional subspace, and then maximize the agreement of embeddings from the Laplacians calculated in the subspaces. To this effect, we propose the following model:

$$\begin{aligned} \max_{\text{All } U_v, W_v} \sum_{v=1}^{m} tr(U_v^T L_v U_v) + \lambda_1 \sum_{v \neq w} R(U_v, U_w), \\ s.t.\ U_v^T U_v = I, W_v^T W_v = I \\ L_v = D_v^{-\frac{1}{2}} K_{W_v^T X_v} D_v^{-\frac{1}{2}} \end{aligned} \tag{2}$$

where $W_v \in \mathbb{R}^{d_v \times k}$ is the projection matrix, $K_{W_v^T X_v}$ is the kernel matrix on the projected subspace, D_v is the degree matrix of $K_{W_v^T X_v}$ and L_v is the corresponding normalized graph Laplacian with $v = 1, \ldots, m$.

The model in (2) seeks a low-dimensional subspace for each view, where the clustering structures are strong (as described by the first term) and the dependence between these clustering embeddings (second term) is maximized. By integrating dimensionality reduction into multi-view clustering, we can find useful information in the projected subspace and maximize the agreement of clusterings there.

However, when there are confounders in the conflicting views, only searching for agreement of spectral clustering in the reduced subspace is not enough. In such cases, the low-dimensional subspace will still be affected by confounders even after the dimensionality reduction.

To tackle this problem, assuming we have prior knowledge that the first view has more discriminatory power with respect to the samples for users' interest, we try to find two non-redundant alternative clustering embeddings U_v and U_v' for each of all the other views. In addition to the regularizer that measures the agreement between desired

embeddings, we introduce another regularizer for confounding correction. We propose our MvKDR model as:

$$\begin{aligned} \max_{\text{All } U,W} \sum_{v=1}^{m} tr(U_v^T L_v U_v) + \sum_{v=1}^{m} tr(U_v'^T L_v' U_v') \\ +\lambda_1 \sum_{v \neq w} R(U_v, U_w) - \lambda_2 \sum_{v,w} R(U_v, U_w'), \\ s.t.\ U^T U = I, W^T W = I, \\ L = D^{-\frac{1}{2}} K_{W^T X_v} D^{-\frac{1}{2}} \end{aligned} \tag{3}$$

where $U \in \{U_v, U_v'\}$, $W \in \{W_v, W_v'\}$, $L \in \{L_v, L_v'\}$, and $D \in \{D_v, D_v'\}$ for $v = 1, \ldots, m$. W_v, U_v and L_v are, respectively, the projection, embedding and Laplacian matrix corresponding to the desired clustering in view v, and W_v', U_v' and L_v' are the projection, embedding and Laplacian matrix corresponding to alternative clustering in view v, respectively.

The model in (3) corrects for confounders at the clustering stage which, as discussed before, fits well in many real applications. It helps find strong clustering structures, through dimensionality reduction, in the consensus embedding U_v (first term in the equation) and in the alternative embedding U_v' (second term). In addition, it maximizes the agreement/dependence between desired embeddings U_v and U_w from different views (third term), and minimizes the agreement/dependence between consensus embedding U_v and alternative embedding U_w' in the other views to correct for confounders (fourth term).

The optimization problem in (3) can be solved by the technique of alternating optimization and kernel dimensionality reduction, which is discussed in Sect. 2.3.

2.3 Optimization algorithm

In this section, we propose an algorithm to solve the optimization problem in (3). We take the alternating maximization strategy in the same way as in co-regularized spectral clustering [3].

We first optimize for U_v by assuming all other variables fixed. For each U_v, the optimization problem of (3) becomes that of (4):

$$\begin{aligned} \max_{U_v} tr(U_v^T (L_v + \lambda_1 \sum_{v \neq w} U_w U_w^T - \lambda_2 \sum_{w} U_w' U_w'^T) U_v) \\ s.t.\ U_v^T U_v = I \end{aligned} \tag{4}$$

The objective function in (4) is the same as the one in spectral clustering with a modified Laplacian matrix. We optimize each U_v with (4) by using eigenvalue decomposition. U_v' can be solved in the same way.

We then optimize for W_v by assuming that all other variables are fixed. Note that the optimization for each W_v and W_v' is independent with U_v and U_v' fixed. Then, with

all but one W_v fixed, the optimization problem becomes:

$$\max_{W_v} tr(D_v^{-\frac{1}{2}} U_v U_v^T D_v^{-\frac{1}{2}} K_{W_v^T X_v}) \quad s.t.\ W_v^T W_v = I, \tag{5}$$

This optimization problem can be solved by gradient ascent. For simplification, we assume that D_v and D'_v are also fixed. Otherwise we can treat $D_v^{-\frac{1}{2}} K_v D_v^{-\frac{1}{2}}$ as the kernel function and apply the chain rule to get the gradient. In practice, we found that this strategy yields similar results compared to fixed D_v and D'_v.

We use the *kernel dimensionality reduction* (KDR) technique [12,13] to solve the problem in (5) with an input kernel matrix $G = D_v^{-\frac{1}{2}} U_v U_v^T D_v^{-\frac{1}{2}}$. Following the scheme of gradient ascent, in each step we calculate the derivative of (5) with a fixed kernel function, i.e. Gaussian kernel. We describe KDR in more detail later and give an example of performing KDR with a Gaussian kernel.

Finally, we repeat these two steps alternatively until convergence. We obtain the clustering by performing k-means on the resulting embedding of the first view (the most informative one). Algorithm 1 provides a summary of our approach `MvKDR`.

Algorithm 1: `MvKDR`

Data: $X_1, ..., X_m$, k, λ_1, λ_2
Result: U_1
`// Initialization`
for $v \in 1, ..., m$ **do**
  K_v, D_v, L_v, U_v = SpectralClustering(X_v);
  Update G_v with (5);
  W_v =KDR(X_v, G_v), $W'_v = W_v$;
end
repeat
  `// Step 1: Given` W`, solve` U
  for $v \in 1, ..., m$ **do**
    Update K_v, D_v, L_v with $W_v^T X_v$;
    Update K'_v, D'_v, L'_v with $W_v'^T X_v$;
  end
  for $v \in 1, ..., m$ **do**
    Solve U_v, U'_v with (4);
  end
  `// Step 2: Given` U`, solve` W
  for $v \in 1, ..., m$ **do**
    Update G_v, G'_v with (5);
    W_v =KDR(X_v, G_v), W'_v =KDR(X_v, G'_v);
  end
until *Converge*;

Kernel Dimensionality Reduction We propose to use *kernel dimensionality reduction* (KDR) to solve (5). KDR was first introduced by Fukumizu et al. [12] for the purpose of regression. Given data X and response Y, KDR aims to find the projection of X onto a subspace $W^T X$ that captures the dependency of X on Y in the Reproducing kernel Hilbert space (RKHS) via two semidefinite kernels $K_{W^T X}$ and K_Y.

Wang et al. [13] extended it to the unsupervised case and employed another kernel-based measure of independence, the *Hilbert-Schmidt Independence Criterion* (HSIC) [17]. HSIC is the Hilbert-Schmidt norm of the cross-covariance operator on two random variables. Its empirical estimate is given by $\text{HSIC}(X, Y) = tr(HK_X HK_Y)$, where H is a centering matrix [17]. The objective function of the HSIC version of KDR can be written as:

$$\max_{W^T W = I} tr(GK_{W^T X}), \tag{6}$$

where G is the centralized input kernel matrix. This is exactly the same as our optimization problem in (5).

Equation (6) can be solved by the steepest gradient ascent method with line search. To fulfill the orthogonal constraints, the gradient is projected onto the tangent space after each update.

With a Gaussian kernel, the function is defined as:

$$K(W^T x_i, W^T x_j) = \exp(-\frac{||W^T x_i - W^T x_j||^2}{2\sigma^2}), \tag{7}$$

where x_i is the i_{th} sample of X_v. To simplify the formula, we write $K_{W^T X_v}$ as K. The derivative of (6) is shown in (8), where $z_i = W^T x_i$ is the i_{th} sample in the projected space. With the orthogonality constraint of W, the problem is non-convex. But as shown in [12,13], the gradient based method works well in practice.

$$\begin{aligned} \frac{\partial(tr(GK))}{\partial W} &= \sum_{i,j=1}^{n} G_{i,j} \frac{\partial K_{ij}}{\partial W} \\ &= \sum_{i,j=1}^{n} (-\frac{1}{\sigma^2} G_{i,j} K_{i,j} (x_i - x_j)(x_i - x_j)^T W) \\ &= \sum_{i,j=1}^{n} (-\frac{1}{\sigma^2} G_{i,j} K_{i,j} (x_i - x_j)(z_i - z_j)^T), \end{aligned} \tag{8}$$

Computational complexity The computational runtime complexity of `MvKDR` consists of two parts: $O(n^3)$ in general for eigen-decomposition of the Laplacian matrix and $O(n^2 dkt_1 t_2)$ for the derivative calculation, where n is the number of samples, d is the largest number of features in all views, k is the dimension of the embedding and t_1 and t_2 are the numbers of iterations for the gradient ascent in KDR and outer loops in Algorithm 1 respectively. Both iterations converge fast in our experiments. Therefore the complexity of `MvKDR` is empirically in the same order as the multi-view spectral clustering method `CoReg` [3] when $d \ll n$.

3 Experiments

In this section, we report our empirical clustering results by comparing the proposed method `MvKDR` to a number of baseline methods on both synthetic and real-world multi-view datasets. In addition, we use our method to perform a survival analysis of cancer patients on two genomic datasets.

3.1 Baseline algorithms and setting

To demonstrate how clustering performance can be improved by our proposed approach, we compared `MvKDR` with the following algorithms:

1. **Single view** (`SPV1` and `SPV2`): Consists in running a spectral clustering on each view separately.
2. **Kernel addition** (`KerAdd`): Adds kernels from different views and performs spectral clustering.
3. **Co-regularized spectral clustering** (`CoReg`): Adopts the co-regularization framework to spectral clustering, pairwise version as described in [3].
4. **Multi-view unsupervised feature selection** (`MVUFS`): Integrates sparse unsupervised feature selection and non-negative matrix factorization into a multi-view clustering framework [16].
5. **Multi-view multi-objective spectral clustering** (`Pareto`): Finds multiple alternative cuts across views with multi-objective Pareto optimization [15].
6. **Multiple non-redundant spectral clustering views** (`mSC`): Finds multiple non-redundant clustering solutions on a single view [11].

All the comparison methods need the number of clusters k to be predetermined. We set k to be the true number of clusters when this is known as ground truth. We choose Gaussian RBF kernel for all the methods and fix the parameter σ using the median of pairwise distances of each view [17]. For methods using KDR we project each view to $k-1$ dimensional subspaces. For methods with a regularization parameter, we set it to $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$, and determine the best parameters with the smallest objective function of k-means. For all methods, we initialize k-means with 10 random re-starts and record the average of the objective function for parameter selection and to report the results. For datasets with class labels, we measure the performance of the clustering methods based on the accuracy (ACC) and the normalized mutual information (NMI), which are widely used for evaluating clusterings. Please refer to [18] for detailed definitions. The source code for `Pareto` and `MVUFS` was provided by the authors [15,16]. We implemented all the rest of the methods. The source code of `MvKDR` can be found online[4].

3.2 Datasets

We evaluate the performance of all the above mentioned methods on three synthetic datasets and on two real datasets: UCI and WebKB.

[4] https://github.com/BorgwardtLab/MvKDR

– **Synthetic datasets:** We generated three synthetic datasets containing views with varying degrees of confounders to compare the performance of the methods listed in Sect. 3.1. The data generation process was as following. We randomly drew 300 samples in 2D Euclidean space with three clusters in order, each with 100 samples. Suppose the samples are represented by two 300-dimensional column vectors $[a_1\ a_2]$, where $a_i = [a_{i,1}^T\ a_{i,2}^T\ a_{i,3}^T]^T$ and $a_{i,j} \in \mathbb{R}^{100}$ for feature $i = 1, 2$, and cluster $j = 1, 2, 3$. We generated $[a_{1,j}\ a_{2,j}]$ from three Gaussian distributions to get three clusters. These three clusters form the main clustering of these 300 samples. We randomly drew two more vectors a_3 and a_4 in the same way but with random order of the samples, such that the alternative clusterings by a_3 and a_4 are independent of the main clustering. For these 300 samples with four features, we constructed `View1` with a_1 and αa_3 and, `View2` with a_2 and βa_4. Three synthetic datasets were generated with different α and β representing different amount of conflicting between views.
– **UCI datasets**[5]**:** We chose six UCI benchmark datasets [19] for evaluation as in [15], namely `Hepatitis`, `Iris`, `Wine`, `Glass`, `Ionosphere`, and `Wdbc`. To construct the two views, we divided the features into two disjoint subsets where the first view contains the first half of the features and the second view the contains the remaining features. The divisions are performed on the data after standardization.
– **WebKB datasets**[6]**:** This dataset contains information of webpages from four universities in the US. We obtained a preprocessed dataset from a previous work [20]. Webpages from each university are document-samples, represented as 0/1-valued word vectors (`View 1`) and hyperlinks between documents (`View 2`). These webpages are classified into one of five classes: course, faculty, student, project and staff. We performed clustering of samples from each university as well as on all the samples. Similarly to the processing of documents described in [3], we first reduced the dimensionality of both views to 100 by Latent Semantic Analysis.

3.3 Results

Synthetic datasets The clustering results of NMI/ACC on synthetic datasets are reported in Table 1. The numbers in parentheses indicate the differences between the obtained result and the best single-view clustering (`SPV1`). The result is highlighted in bold if the improvement is at least 0.01.

From Table 1 we can see that the clusterings of `View 1` and `View 2` have little agreement. Our proposed method `MvKDR` improves significantly over `SPV1` and provides the best results in all scenarios with an average improvement in NMI/ACC of $+9.7\%/4.6\%$. `KerAdd`, `Pareto` and `MVUFS` severely degrade the clustering performance compared to `SPV1` in all three datasets, with a reduction of NMI/ACC of $-39.1\%/32.7\%$ for `KerAdd`, $-18.7\%/15.8\%$ for `Pareto` and $-11.6\%/8.4\%$ for `MVUFS`. `mSC` performs almost the same as `SPV1`. `CoReg` can improve the clustering quality on `Syn1`, but shows no benefit in `Syn2`, and its performance degrades severely on `Syn3` ($-38.4\%/29.4\%$).

[5] https://archive.ics.uci.edu/ml/datasets.html
[6] http://linqs.umiacs.umd.edu/projects/projects/lbc/

Table 1: NMI/ACC in % for Synthetic Data and difference to that of SPV1 in parentheses, the result is highlighted in bold if the improvement to SPV1 is more than 1%.

		SPV1	SPV2	KerAdd	CoReg	mSC	Pareto	MVUFS	MvKDR
NMI	Syn1	71.5	0.80	40.6 (-30)	**75.8 (+4.3)**	71.3 (-0.2)	46.1 (-25)	57.9 (-13)	**80.1 (+8.6)**
	Syn2	71.5	0.70	33.3 (-38)	71.8 (+0.3)	71.3 (-0.2)	52.7 (-18)	57.9 (-13)	**74.6 (+3.1)**
	Syn3	53.9	0.70	5.60 (-48)	15.5 (-38)	54.2 (+0.3)	49.0 (-4.9)	45.5 (-8.4)	**71.5 (+17)**
ACC	Syn1	91.9	38.2	62.9 (-29)	**93.2 (+1.3)**	91.7 (-0.2)	72.9 (-19)	82.8 (-9.1)	**94.7 (+2.8)**
	Syn2	91.9	37.8	60.1 (-31)	91.9 (+0.0)	91.7 (-0.2)	76.6 (-15)	82.8 (-9.1)	**93.0 (+1.1)**
	Syn3	81.5	37.8	44.2 (-37)	52.1 (-29)	81.8 (+0.4)	74.6 (-6.9)	74.3 (-7.2)	**91.3 (+9.8)**

UCI dataset Table 2 shows the clustering results on the UCI datasets and, in parentheses, the difference of NMI/ACC to the result of the best single-view clustering (SPV1). As before, the result is highlighted in bold if the improvement is at least 0.01. The Hepa and Iono datasets stand for Hepatitis and Ionosphere respectively.

From Table 2 we can see that the clusterings on the two constructed views for Hepatitis, Iris and Wdbc are not in agreement as the results of spectral clustering on View 1 are much closer to the ground truth than those on View 2. For Wine and Glass we see the opposite behavior: the clusterings on each view partly reflect the ground truth. For Ionosphere it is hard to draw any conclusion because clustering on each view performs badly.

MvKDR improves the best single-view clustering on all six datasets, whether there are conflicting views or not, with an average improvement in NMI/ACC of $+6.5\%/2.7\%$. In addition, MvKDR provides the best results on one dataset, second best results on two datasets, and third best results on the other three datasets.

CoReg can improve the clustering quality on four out of six datasets since many of them have clusterings that agree across views, with an average improvement in NMI/ACC of $+4.8\%/2.3\%$, which is inferior to the one obtained with MvKDR. In addition, it cannot gain from the second view on Iris and its performance degrades on Hepatitis.

The clustering performance of KerAdd is also degraded on datasets with conflicting views such as Hepatitis, Iris and Wdbc, with an average degradation of NMI/ACC of $-27.8\%/6.3\%$.

Pareto provides the best results on the Iris dataset, but its performance degrades severely on the other five datasets with an average degradation in NMI/ACC of $-20.0\%/11.1\%$. The reason might be that Pareto only considers binary alternative cuts. It is not clear how to merge these binary cuts into higher number of clusters.

MVUFS also provides the best results on Ionosphere, but is not effective on the rest of datasets with an average degradation in NMI/ACC of $-27.8\%/15.3\%$. This may be due to the fact that the dimensionality of the UCI datasets is rather small, and this affects the method as it is based on feature selection.

As mentioned before, mSC tries to find multiple clustering solutions of a single view. It is clear from the table that mSC improves over SPV1 on Glass and Wdbc by

Table 2: NMI/ACC in % for UCI Data and difference to that of SPV1 in parentheses, the result is highlighted in bold if the improvement to SPV1 is more than 1%.

		SPV1	SPV2	KerAdd	CoReg	mSC	Pareto	MVUFS	MvKDR
NMI	Hepa	25.9	7.40	19.3 (-6.6)	25.0 (-0.9)	21.8 (-4.1)	5.70 (-20)	25.1 (-0.8)	26.6 (+0.7)
	Iris	67.3	10.7	32.9 (-34)	67.3 (+0.0)	67.7 (+0.4)	**80.8 (+13)**	3.80 (-63)	**75.8 (+8.5)**
	Wine	75.7	54.7	**88.4 (+12)**	**91.1 (+15)**	60.6 (-15)	42.4 (-33)	9.40 (-66)	**88.5 (+12)**
	Glass	37.9	25.2	**43.1 (+5.2)**	**43.4 (+5.5)**	**44.0 (+6.1)**	11.0 (-26)	24.1 (-13)	**43.7 (+5.8)**
	Iono	9.50	9.60	**12.0 (+2.5)**	**11.7 (+2.2)**	10.0 (+0.5)	4.60 (-4.9)	**16.8 (+7.3)**	**11.7 (+2.2)**
	Wdbc	51.3	3.50	49.9 (-1.4)	**57.9 (+6.6)**	**58.6 (+7.3)**	2.80 (-48)	46.5 (-4.8)	**57.3 (+6.0)**
ACC	Hepa	77.4	61.9	72.9 (-4.5)	76.6 (-0.8)	71.0 (-6.4)	58.9 (-18)	76.9 (-0.5)	78.1 (+0.7)
	Iris	94.0	69.0	82.0 (-12)	94.0 (+0.0)	94.0 (+0.0)	**97.0 (+3.0)**	51.0 (-43)	**96.0 (+2.0)**
	Wine	92.1	84.3	**97.2 (+4.9)**	**97.8 (+5.7)**	83.4 (-8.7)	70.8 (-21)	49.7 (-42)	**96.9 (+4.7)**
	Glass	71.5	56.5	**76.2 (+4.7)**	**76.6 (+5.5)**	**78.5 (+7.0)**	71.5 (+0.0)	59.3 (-12)	**75.7 (+4.2)**
	Iono	68.4	69.2	70.1 (+0.9)	69.8 (+0.6)	**69.5 (+1.5)**	65.0 (-3.4)	**75.1 (+6.7)**	69.8 (+0.6)
	Wdbc	89.6	63.3	87.2 (-2.4)	**91.7 (+2.1)**	**92.1 (+2.5)**	63.1 (-26)	87.9 (-1.7)	**91.7 (+2.1)**

only correcting possible confounders in a single view. This is a sign that correction of confounders may indeed improve the clustering. However, mSC performs much worse on the rest of the datasets because it only uses single view information.

Figure 1 depicts the mean difference of NMI values of different methods with regard to the best-performing technique on each dataset (i.e. the largest NMI value obtained in any run). Similar observations can be seen for the ACC value, which is not shown due to the space limitations. From the figure it is clear that MvKDR outperforms all other methods on the UCI datasets.

WebKB dataset Table 3 depicts the clustering results on the WebKB datasets. Cor, Tex, Was and Wis stand for the dataset from University of Cornell, Texas, Washington and Wisconsin. As it was the case for the UCI datasets, the improvement/degradation compared to SPV1 is shown in parentheses, numbers are highlighted in bold if the improvement is at least 0.01.

In Table 3 we see that the clustering on the link view (View 2) hardly agrees with the ground truth labels (on average, NMI of only 8.5%). The proposed method MvKDR improves the best single-view clustering on all but one dataset and provides the best results on two datasets and second best on the other two, with an average improvement in NMI/ACC of $+1.4\%/0.6\%$. KerAdd performs poorly on four out of five datasets compared to the best single view, with an average of degradation in NMI/ACC of $-5.4\%/1.8\%$. CoReg outputs nearly the same results as the best single view clustering on four datasets and has worse performance on the other one. The clustering performance of Pareto degrades on all five datasets (average degradation in NMI/ACC of $-17.3\%/12.2\%$), again the reason is probably that it only considers alternative binary cuts and there are five clusters in WebKB datasets. The performance of MVUFS also degrades significantly on four out of five WebKB datasets (degradation in NMI/ACC of

Table 3: NMI/ACC in % for Webkb Data and difference to that of SPV1 in parentheses, the result is highlighted in bold if the improvement to `SPV1` is more than 1%.

		SPV1	SPV2	KerAdd	CoReg	mSC	Pareto	MVUFS	MvKDR
NMI	Cor	41.4	6.60	27.0 (-14)	41.7 (+0.3)	40.0 (-1.)	14.1 (-27)	11.8 (-29)	**45.1 (+3.7)**
	Tex	36.7	16.0	24.2 (-12)	25.6 (-11)	**38.2 (+1.5)**	11.5 (-25)	27.7 (-9.0)	**37.9 (+1.2)**
	Was	34.9	7.80	33.4 (-1.5)	35.0 (+0.1)	32.9 (-2.0)	27.7 (-7.2)	20.7 (-14)	34.6 (-0.3)
	Wis	33.0	6.50	**40.2 (+7.2)**	33.0 (+0.0)	34.4 (+1.4)	20.1 (-12)	6.30 (-26)	33.8 (+0.8)
	All	16.5	5.70	10.7 (-5.8)	16.8 (+0.3)	15.9 (-0.6)	2.70 (-13)	11.6 (-4.9)	**17.9 (+1.4)**
ACC	Cor	55.9	38.5	46.0 (-9.9)	54.4 (-1.5)	53.8 (-2.1)	38.5 (-17)	37.9 (-18)	**57.2 (+1.3)**
	Tex	46.6	49.9	49.5 (-0.4)	47.1 (-2.8)	48.3 (-1.6)	35.9 (-10)	**53.4 (+3.5)**	48.0 (-1.9)
	Was	50.5	39.1	48.8 (-0.7)	50.6 (+0.1)	49.3 (-1.2)	27.7 (-22)	42.3 (-8.2)	50.5 (+0.0)
	Wis	50.9	37.4	**59.9 (+9.9)**	50.9 (+0.0)	**52.0 (+1.1)**	41.0 (-9.9)	40.1 (-10)	**53.3 (+2.4)**
	All	41.9	28.3	34.7 (-7.9)	41.7 (-0.2)	41.3 (-0.6)	29.5 (-12)	38.9 (-3.0)	**43.2 (+1.3)**

$-17.3\%/12.2\%$). The reason might be the existence of confounders in the conflicting views.

The overall NMI performance of different methods with regard to the best-performing technique on WebKB datasets is also shown in Fig. 1. `MvKDR` outperforms all other methods and is the only method to improve the clustering only on `View 1`. These experiments show the effectiveness of our proposed method `MvKDR` when clustering with conflicting views.

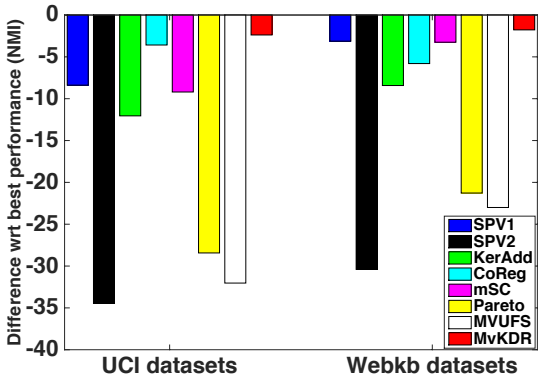


Fig. 1: The mean difference of NMI of different methods with respect to the best-performing technique on each dataset, grouped by two cases: UCI and WebKB datasets.

3.4 Cancer patients survival analysis

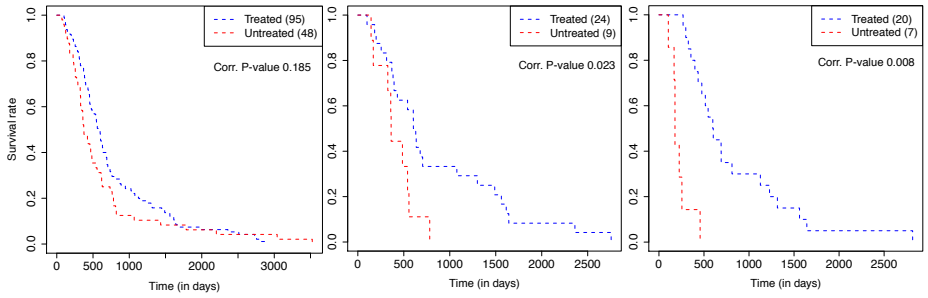
We conducted additional analyses of cancer genomics data from The Cancer Genome Atlas (TCGA) Research Network [21]. The data were preprocessed and provided by Wang et al. [22] and it includes five cancer types: glioblastoma multiforme (GBM), breast invasive carcinoma (BRCA), kidney renal clear cell carcinoma (KIRC), colon adenocarcinoma (COAD) and lung squamous cell carcinoma (LUSC). For each type of cancer, we used gene expression and DNA methylation expression data as two separate views for clustering. Our goal is to identify clusters in which patients can be considered to have a specific cancer subtype. This is a discovery process as there are no ground truth labels. Yet, we have information about certain drugs that some patients took. We performed a survival analysis on the stratified clusters from `MvKDR` and from the other methods and investigated if patients within a cluster had the same response to the drug treatment (here response is measured as survival time). We expect to find clusters of patients that respond the same to a drug treatment. We performed a two-sample t-test within each cluster to compare whether the survival time is significantly different between the patients that received the drug versus those that did not.

Patients without drug treatment (or with missing drug information) were removed from the analysis. We end up with 141 samples for GBM cancer, 76 samples for BRCA and 0, 20, 27 samples for COAD, KIRC and LUSC respectively. Due to small sample sizes of other three cancer types, we perform the survival analysis only on GBM and BRCA. We select the drug that is used by most of patients in each cancer type, Temozolomide for GBM and Cytoxan for BRCA. Of the 141 GBM patients, 95 were treated with Temozolomide and 52 out of the 76 BRCA patients were treated with Cytoxan.

To verify our assumption of conflicting views on this dataset, we first conducted a spectral clustering on each view and measured the NMI value between them. The NMI between clustering results from gene expression and DNA methylation is only 0.021 and 0.051 for number of clusters $k = 2$ and $k = 3$, respectively. This experiment shows that the gene expression view and DNA methylation view conflict with each other.

Next we compare the our method `MvKDR` to the baseline method of Single View spectral clustering `SPV1` and `SPV2` and state-of-the-art multi-view spectral clustering method `CoReg`. Gene expression profiles are more often used to define cancer subtypes, therefore we used this view as the more informative one for `CoReg` and `MvKDR`. Similarly to the analyses on the UCI and WebKB datasets, the parameters of `CoReg` and `MvKDR` were determined by the objective function of k-means. We performed clustering with increasing number of clusters k, starting from $k = 2$. For each cluster we conducted a two-sample t-test with different variance of survival time for patients with and without the drug treatment. In our analysis we found that when $k > 6$ the number of samples in a cluster is too small and the clustering tends to be more correlated to the previous one with smaller k.

`MvKDR` detects two significant clusters in regards to the survival analysis of patients treated with the drug Temozolomide versus those not treated with it. Figure 2 depicts the comparison of survival time for all patients and of patients in the two discovered clusters. In cluster 0 with $k = 6$, patients with the drug treatment had a significantly increased survival time with a p-value < 0.01 after Bonferroni correction. In cluster 1 with $k = 2$ we make a similar observation that treated patients live longer than untreated



(a) All Samples (b) $k = 2$, Cluster 1 (c) $k = 6$, Cluster 0

Fig. 2: Survival analysis of GBM patients for treatment with Temozolomide in the whole dataset and two significant clusters found by MvKDR. The numbers in parentheses denote the number of patients in the group; the p-values are corrected for multiple testing using the Bonferroni method.

ones with a p-value < 0.05 after Bonferroni correction. For patients in the whole GBM dataset and in other clusters found by MvKDR, we did not observe significant differences of survival time between treated and untreated patients. This experiment shows that MvKDR can discover meaningful subgroups of patients based on their genomic profiles, where the drug treatment of Temozolomide can significantly increase the survival time. These findings will be useful for recommendations of Temozolomide treatment to patients with genomic profiles similar to those found by our model.

We further performed spectral clustering on each data view and CoReg on both views, where SPV1 corresponds to the gene expression view and SPV2 to the DNA methylation view. Both SPV1 and CoReg found one significant cluster regarding survival time of treated and untreated patients of the drug after Bonferroni corection. Figure 3 depicts the number of overlapping patients of all four significant clusters found by SPV1, CoReg and MvKDR. From the figure we can see that Cluster 2 found by MvKDR, Cluster 3 found by SPV1 and Cluster 4 found by CoReg are essentially identical, with the majority of the patients overlapping. Further, Cluster 2 found by MvKDR achieves the smallest p-value compared to the other two. Cluster 1 found by MvKDR overlaps with the other three in only four patients. This analysis shows that MvKDR is able to find useful clusters that can be also detected by state-of-the-art methods. In addition, MvKDR discovers a novel cluster that was missed by the comparison methods. One possible reason for this could be that MvKDR unveils the masked information in the DNA methylation view by using kernel dimensionality reduction and confounding correction.

We performed survival analysis on the BRCA dataset as well, where we have similar observations as on GBM dataset. Specifically, MvKDR detects one significant cluster of 51 patients regarding survival analysis of the drug Cytoxan with a p-value of 0.036 after Bonferroni correction. SPV2 finds one significant cluster of 50 patients with p-value of 0.005. The number of overlap of patients between two groups is 33. This shows again that MvKDR can detect meaningful and novel cluster compared to the baseline algorithms.

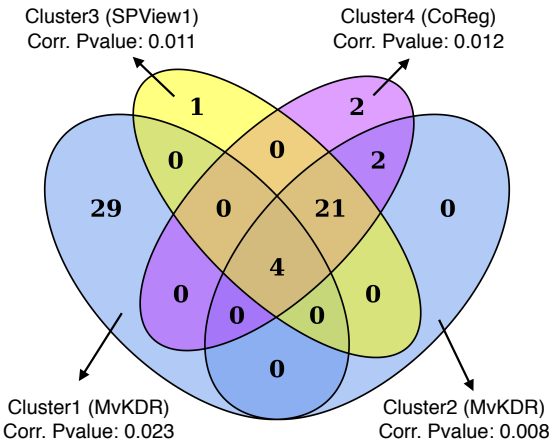


Fig. 3: Overlapping comparison of all four significant clusters of survival analysis on GBM data from different methods. `MvKDR` detects Cluster 2 that is almost the same as Cluster 3 (`SPV1`) and Cluster 4 (`CoReg`) with a smaller P-value of 0.008. In addition, `MvKDR` finds a novel significant cluster (Cluster 1) missed by other methods.

4 Conclusion

Most existing multi-view learning approaches suffer on conflicting views and confounders. In this work, we propose a new approach named `MvKDR` to find the desired consensus clustering across different views, which is normally hidden or masked by confounders in conflicting views. With prior knowledge about the most informative view, our main idea is to extract two kinds of independent information from each of all the other views: the first is consistent with the desired consensus structure and the second is independent of it. The consensus clustering can be obtained by the consistent information across all the views. Our experiments on synthetic and real datasets show that the `MvKDR` significantly improves the clustering. In our model, we assume that we have prior knowledge about the most informative view, which in certain cases may be difficult to obtain in practice. If so, it is challenging to distinguish which clustering is more interesting. The problem may be solved in a semi-supervised manner with a few user inputs as constraints that guide the direction of the dimensionality reduction. Extending our approach in this direction may be a topic of future work.

5 Acknowledgments

This work was funded in part by the SNSF Starting Grant Significant Pattern Mining (XH, KB), the Marie Curie Initial Training Network MLPM2012, Grant No. 316861 (KB), the NSFC projects 11471256 and 11631012 (LL).

The authors greatly acknowledge Dean Bodenham for helpful discussions and proofreading of the manuscript.

References

1. Bickel, S., Scheffer, T.: Multi-view clustering. In: ICDM. (2004) 19–26
2. Chaudhuri, K., Kakade, S.M., Livescu, K., Sridharan, K.: Multi-view clustering via canonical correlation analysis. In: ICML. (2009) 129–136
3. Kumar, A., Rai, P., III, H.D.: Co-regularized multi-view spectral clustering. In: NIPS. (2011) 1413–1421
4. Kumar, A., III, H.D.: A co-training approach for multi-view spectral clustering. In: ICML. (2011) 393–400
5. Xia, R., Pan, Y., Du, L., Yin, J.: Robust multi-view spectral clustering via low-rank and sparse decomposition. In: AAAI. (2014) 2149–2155
6. Tang, J., Hu, X., Gao, H., Liu, H.: Unsupervised feature selection for multi-view data in social media. In: SDM. (2013) 270–278
7. Wang, H., Nie, F., Huang, H.: Multi-view clustering and feature learning via structured sparsity. In: ICML. (2013) 352–360
8. Gao, J., Han, J., Liu, J., Wang, C.: Multi-view clustering via joint nonnegative matrix factorization. In: SDM. (2013) 252–260
9. Cui, Y., Fern, X.Z., Dy, J.G.: Non-redundant multi-view clustering via orthogonalization. In: ICDM. (2007) 133–142
10. Gondek, D., Hofmann, T.: Non-redundant data clustering. In: ICDM. (2004) 75–82
11. Niu, D., Dy, J.G., Jordan, M.I.: Multiple non-redundant spectral clustering views. In: ICML. (2010) 831–838
12. Kenji Fukumizu, F.R.B., Jordan, M.I.: Kernel dimension reduction in regression. The Annals of Statistics **37**(4) (2009) 1871–1905
13. Wang, M., Sha, F., Jordan, M.I.: Unsupervised kernel dimension reduction. In: NIPS. (2010) 2379–2387
14. Christoudias, C.M., Urtasun, R., Darrell, T.: Multi-view learning in the presence of view disagreement. In: Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence. UAI (2008) 88–96
15. Davidson, I., Qian, B., Wang, X., Ye, J.: Multi-objective multi-view spectral clustering via pareto optimization. In: SDM. (2013) 234–242
16. Qian, M., Zhai, C.: Unsupervised feature selection for multi-view clustering on text-image web news data. In: CIKM. (2014) 1963–1966
17. Gretton, A., Bousquet, O., Smola, A.J., Schölkopf, B.: Measuring statistical dependence with hilbert-schmidt norms. In: ALT. (2005) 63–77
18. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: SIGIR. (2003) 267–273
19. Lichman, M.: UCI machine learning repository (2013)
20. Sen, P., Namata, G.M., Bilgic, M., Getoor, L., Gallagher, B., Eliassi-Rad, T.: Collective classification in network data. AI Magazine **29**(3) (2008) 93–106
21. Network, T.C.G.A.: The cancer genome atlas. http://cancergenome.nih.gov/ (2006)
22. Wang, B., Mezlini, A.M., Demir, F., Fiume, M., Tu, Z., Brudno, M., Haibe-Kains, B., Goldenberg, A.: Similarity network fusion for aggregating data types on a genomic scale. Nat Meth **11**(3) (March 2014) 333–337