Constrained Spectral Clustering with Distance Metric Learning

Yuanli Pei
School of EECS, Oregon State University
Email: peiy@eecs.oregonstate.edu

Teresa Vania Tjahja
School of EECS, Oregon State University
Email: tjahjat@eecs.oregonstate.edu

Abstract—Spectral clustering is a flexible clustering technique that finds data clusters in the spectral embedding space of the data. It doesn’t assume convexity of the shape of clusters, and is able to find non-linear cluster boundaries. Constrained spectral clustering aims at incorporating user-defined pairwise constraints in to spectral clustering. Typically, there are two kinds of pairwise constraints, Must-Link constraint and Cannot-Link constraint. These constraints represent prior knowledge indicating whether two data objects should be in the same cluster or not; thereby aiding clustering. In this paper, we propose a novel approach that alternatively learns a distance metric from such constraints, and finds the clustering solution on the spectral embedding of the transformed data by the learned metric. Although our formulation is non-convex, empirically we observe the alternative optimization can effectively find the local optimum. We show that the proposed method outperforms both existing methods that only learns distance metric from constraints, and constrained spectral clustering method without distance metric learning.

I. INTRODUCTION

Clustering is the task of grouping objects into clusters such that similar objects are grouped together. Among various techniques, spectral clustering [9] is one of the most flexible methods that doesn’t make assumptions on cluster shapes such as convexity. Due to this property, spectral clustering is applicable to a variety of data types, and has drawn much attention recently.

Traditionally, clustering is viewed as an unsupervised method, where human provided information is not involved in the clustering process. However, in some cases, side information about the problem domain is available to aid clustering beyond instances themselves. Such information is usually encoded as constraints. One common and useful type of constraints is in the form of pairwise constraints [12], [14], [3], [2], where each Must-Link (ML) constraint specifies that a pair of instances belong to the same cluster, and each Cannot-Link (CL) constraint indicates that two instances belong to different clusters.

In this paper, we consider the problem of spectral clustering with pairwise constraints, where the goal is to find a spectral clustering solution that respects the given constraints. We propose a novel approach that alternate between learning a distance metric from the pairwise constraints and finding the spectral clustering solution on the transformed metric space. We encode first the pairwise constraints as distance constraints, and then relax the constraints as a loss minimization problem. Although the problem is non-convex, we empirically observe that the alternative optimization converges in very few iterations. We evaluate our method on benchmark machine learning datasets and experiment results demonstrate its effectiveness of utilizing pairwise constraints to aid clustering.

II. RELATED WORK

Distance Metric Learning with Pairwise Constraints: Most of the existing work in metric learning relies on learning a Mahalanobis distance. For example, earlier work by [14] uses a semidefinite programming formulation under similarity (ML) and dissimilarity (CL) constraints. The work in [5] expresses the problem of learning a Mahalanobis distance from pairwise constraints as a particular Bregman optimization problem of minimizing the LogDet divergence subject to linear constraints. In general, these methods are usually followed by the Kmeans clustering to find cluster boundaries. None of them are designed specifically for spectral clustering which is suitable for finding non-linear decision boundary. In this paper, we design the distance metric learning technique for spectral clustering.

Constrained Spectral Clustering: The first approach to integrate constraints into spectral clustering was based on the idea of modifying the weight matrix to enforce the ML and CL constraints and then solving the resulting unconstrained problem [7]. Another approach is to adapt the embedding obtained from the first $K$ eigenvectors of the graph Laplacian [8]. Closer to the original normalized graph cut problem [11] (the original spectral clustering problem) are the approaches that add ML and CL constraints as penalty terms in the objective [17], [6], [15], [13]. Our method differs from these methods in the senses that we learn the distance metric from the constraints, and find the spectral clustering solutions from the transformed space, instead of directly forming spectral embedding of data in the original feature space.

III. PROBLEM STATEMENT

Formally, let $X = \{x_1,\ldots, x_N\}, x_i \in \mathbb{R}^d$ be the given data, where $d$ is the dimension of the feature space. Let $\mathcal{M} = \{(i,j)\}$ be the given set of ML constraints, and let $\mathcal{C} = \{(i,j)\}$ be the given set of CL constraints. A pair $(i,j) \in \mathcal{M}$ indicates that $x_i$ and $x_j$ belong to the same cluster, and $(i,j) \in \mathcal{C}$ means that $x_i$ and $x_j$ belong to different clusters. Let $W \in \mathbb{R}^{d \times q}$ be the linear transformation matrix that corresponds to a Mahalanobis distance, where $q$ is the number of the transformed dimensions. The squared Mahalanobis distance of $x_i$ and $x_j$ based on $W$ is given by

$$d_W(i,j) = \|W^T x_i - W^T x_j\|_2^2$$

$$= (x_i - x_j)^T W W^T (x_i - x_j). \quad (1)$$
Our goal is to learn the distance metric $W$ and find the spectral clustering solution from the transformed data by the distance metric.

In the following, we assume that the number of clusters $K$ is given. The discussion of selecting $K$ is beyond the purpose of this paper.

### IV. Methodology

#### A. Preliminaries on Spectral Clustering

First we briefly review the spectral clustering. Let $A = [a_1, \cdots, a_K]$ be a partition matrix, where each column $a_k$ is a binary assignment vector for cluster $X_k$, with $a_{ik} = 1$ if instance $x_i$ is assigned to cluster $X_k$ and 0 otherwise. Let $S$ be the given symmetric similarity matrix of instances, with $s_{ij}$ representing the similarity between instance $x_i$ and $x_j$. The similarity is usually computed using a kernel function, e.g. Gaussian kernel, in which case it is calculated as

$$s_{ij} = \exp \left( -\frac{||x_i - x_j||^2_2}{2\sigma^2} \right),$$

where $\sigma$ is the bandwidth. Let the degree matrix $D = \text{Diag}(S1_K)$, where $\text{Diag}()$ forms a diagonal matrix using the input vector. $1_N$ denotes a $N$-dimensional vector of all 1’s. The goal of spectral clustering is to minimize the between group similarity and meanwhile maximize the within group similarity. There are several objective functions that capture this goal. Here we focus on the Normalized LinkRatio $K$-way partition objective [16]. The problem of spectral clustering is defined as

$$\max_A \frac{1}{K} \sum_{k=1}^K \frac{a_k^T S a_k}{a_k^T D a_k},$$

s.t. $A \in \{0, 1\}^{N \times K}$, $A1_K = 1_N$,

where the constraints ensure that each instance is only assigned to one cluster.

The objective function can be reformulated as

$$\frac{1}{K} \sum_{k=1}^K \frac{a_k^T S a_k}{a_k^T D a_k} = \frac{1}{K} \sum_{k=1}^K a_k^T D^{1/2} D^{-1/2} S D^{-1/2} D^{1/2} a_k.$$

Now define $U = [u_1, \cdots, u_K]$, with $u_k = \frac{D^{1/2} a_k}{||D^{1/2} a_k||}$. Ignoring the discrete constraint for $U$ at this stage, one can formulate a new clustering problem with respect to the variable $U$ as

$$\max_U \quad \text{tr}(U^T D^{1/2} S D^{-1/2} U)$$

s.t. $U^T U = I$ \hspace{1cm} (5)

where the constraint comes naturally from the definition of $U$. Although the new problem (5) is not convex due to the orthonormal constraint in $U$, it has an analytical solution, namely, the optimal $U$ is the eigenvectors associated with the $K$ largest eigenvalues of $D^{-1/2} S D^{-1/2}$ [4]. These eigenvectors are viewed as the spectral embedding of the original data. Correspondingly, a discrete solution $A$ of the original problem can be obtained by taking a rounding procedure from $U$ (e.g. using K-means or the approach proposed in [16]).

#### B. Formulation

1) Initial Formulation: Recall that our goal is to learn the transformation $W$ and find the data partition matrix $U$. Using the Gaussian kernel for spectral clustering, the similarity of $x_i$ and $x_j$ in the new metric space becomes

$$s_{ij} = \exp \left( -\frac{||W^T x_i - W^T x_j||^2_2}{2\sigma^2} \right)$$

To encode the pairwise constraints, for each $(i, j) \in M$, we restrict that their similarity to be greater than a pre-specified value $s_0$, i.e., $s_{ij} \geq s_0$. Similarly, for each $(i, j) \in C$, we restrict that their similarity to be less than such prior, namely, $s_{ij} \leq s_0$. For identifiability reasons, we also regularize the parameter $W$ using the squared L-2 norm. Summing up together, our formulation becomes

$$\min_{U, W} \quad -\text{tr}(U^T D^{-1/2} S D^{-1/2} U) + \lambda ||W||^2_2$$

s.t. $s_{ij} = \exp \left( -\frac{||W^T x_i - W^T x_j||^2_2}{2\sigma^2} \right), \quad \forall i, j \hspace{1cm} (7)$

$$s_{ij} \geq s_0, \quad \forall (i, j) \in M$$

$$s_{ij} \leq s_0, \quad \forall (i, j) \in C$$

$$U^T U = I, \quad D = \text{Diag}(S1_K)$$

where $\lambda$ is a balancing parameter.

The inequality constraints in problem (7) are non-linear in the variable $U$. After rearrangement of the terms, we can rewrite the inequality constraints in to linear form and have
the following equivalent problem:

\[
\min_{U,W} -\text{tr}(U^TD^{-1/2}SD^{-1/2}U) + \lambda \|W\|_2^2 \\
\text{s.t. } s_{ij} = \exp \left( -\frac{d_W(i,j)}{2\sigma^2} \right), \quad \forall i,j \\
d_W(i,j) \leq d_0, \quad \forall (i,j) \in \mathcal{M} \\
d_W(i,j) \geq d_0, \quad \forall (i,j) \in \mathcal{C} \\
U^TU = I, \quad D = \text{Diag}(S1_N)
\]

where \(d_W(i,j)\) is defined in Eq. (1) and \(d_0 = -2\sigma^2 \log s_0\).

2) Constraint Relaxation by Loss Minimization: In reality, due to the complex structures of the data, the pairwise constraints may not be all satisfied. To deal with this, we allow some constraints to be violated but enforce a loss for such violations. Our goal is to minimize the total loss incurred by all the violations.

We now describe how to penalize such violation of constraints. First define \(t_{ij}\) as the label of the pairwise constraint \((i,j)\), namely,

\[
t_{ij} = \begin{cases} 
+1, & (i,j) \in \mathcal{M} \\
-1, & (i,j) \in \mathcal{C}
\end{cases}
\]

For each pairwise constraints \((i,j)\), we define the following loss function

\[
l_{ij} = \max\{0, t_{ij}(d_W(i,j) - d_0)\}, \quad \forall (i,j) \in \mathcal{M} \cup \mathcal{C}.
\]

The loss function means that, if the pairwise constraint for \((i,j)\) in problem (8) is satisfied, then there is no penalty enforced. Otherwise, an amount of \(|d_W(i,j) - d_0|\) loss is incurred. An example of the loss as a function of \(d_W(i,j)\) is shown in Figure 2.

Overall, we want to minimize the losses such that all the constraints are satisfied. Now instead of solving the problem (8), we aim to solve the relaxed problem with the goal of loss minimization:

\[
\min_{U,W} -\text{tr}(U^TD^{-1/2}SD^{-1/2}U) + \alpha \sum_{(i,j) \in \mathcal{M} \cup \mathcal{C}} l_{ij} + \lambda \|W\|_2^2 \\
\text{s.t. } s_{ij} = \exp \left( -\frac{d_W(i,j)}{2\sigma^2} \right), \quad \forall i,j \\
U^TU = I, \quad D = \text{Diag}(S1_N)
\]

where \(\alpha\) is a trade-off parameter for the penalty term, and \(M = |\mathcal{M} \cup \mathcal{C}|\) is the total number of ML and CL constraints that serves as a normalization factor.

V. OPTIMIZATION

A. Initial Attempt with CVX

Our initial attempt at solving the problem was through the use of CVX toolbox in MATLAB. First, we formulate our problem for CVX as shown in (9). However, it could not be solved by CVX since the first term \(-\text{tr}(U^TD^{-1/2}SD^{-1/2}U)\) is not convex in \(W\), and the second term \(\sum_{(i,j) \in \mathcal{M} \cup \mathcal{C}} l_{ij}\) is non-convex in \(W\) when CL constraints are considered. These properties violate the Disciplined Convex Programming (DCP) rule set defined by CVX. We then went back to use problem (8), which was also rejected by CVX since it only accepts scalar quadratic forms such as

\[(Ax - b)^TQ(Ax - b),\]

where \((Ax - b)\) is a vector. Thus, we need to find another approach to solve our optimization problem.

B. Alternative Optimization Algorithm

Now we aim to solve for the non-convex optimization problem (9). We propose an alternative optimization approach similar with that used in [10]. The general optimization steps are as follows:

We first randomly initialize \(W^0\), and then alternate between the following two steps until convergence (the change of the objective is sufficiently small).

- For a fixed \(W\), solve for \(U\): When \(W\) is fixed, the problem becomes the same with the standard spectral clustering in Eq. (5) and there is a closed-form solution as described in Section IV-A. Namely, we can solve the problem using eigen-decomposition.

- For a fixed \(U\), solve for \(W\): When \(U\) is fixed, we don’t have the \(U^TU = I\) constraint in the problem (9). If we substitute other equality constraints to the objective, then the problem become unconstrained in \(W\). We can use any descent method to find a local minimum in variable \(W\). In our experiment, we use the MATLAB \texttt{fminunc} that calls L-BFGS method with BTLS to solve for \(W\).

C. Implementation Details

1) Solve \(W\): Gradient Correctness: In the step of solving for \(W\), the L-BFGS method requires the computation of the gradient for the objective. The first and the third term in our objective are differentiable, and the gradient can be easily computed. However, the second term is non-differentiable due to the fact that each \(l_{ij}\) is non-differentiable at the point \(d_W(i,j) = d_0\), as can be seen from Figure 2. Here, we use the sub-gradient instead.

The subgradient of each \(l_{ij}\) w.r.t. \(W\) is

\[
\frac{\partial l_{ij}}{\partial W} = \begin{cases} 
  t_{ij} \frac{\partial d_W(i,j)}{\partial W}, & t_{ij}(d_W(i,j) - d_0) > 0 \\
  0, & \text{o.w.}
\end{cases}
\]
data points from four bivariate Gaussian distributions, with
straints. We generate a synthetic dataset consisting of 100
method in finding clustering solution using pairwise con-
A. A Toy Example

Here use a toy example to illustrate the behavior of our
method in finding clustering solution using pairwise con-
straints. We generate a synthetic dataset consisting of 100
data points from four bivariate Gaussian distributions, with
25 points per Gaussian. We let the points from two of the
Gaussian belong to the “blue” cluster and points from the
other two Gaussian belong to the “red” cluster. The plot of
the dataset is shown in Figure 4(a). We then apply spectral
clustering (SP), CSP (our method) without constraints, and
CSP using 20 randomly generated constraints on the dataset.

Figure 4(b) shows the spectral embedding of the data
formed by SP. We can see that the points that near the
boundaries of the two clusters are mixed together after the
embedding. Thus we couldn’t find a clear-cut decision bound-
ary in the embedded space, although the clusters in original
space are almost separable.

Figure 4(c) and 4(d) show the linear transformed data and
the spectral embedded data when CSP is performed without
using any pairwise constraints. Note that when there is no
constraint provided for CSP, it still differs from SP in the sense
that CSP both learns the linear transformation of the data forms
the spectral embedding, while SP only performs the later. We
see that although there is no supervision information provided,
CSP still learns a reasonable linear transformation of the data.
The spectral embedding of the transformed data can separate
most of the points from the two clusters, with only two blue
points mixed in the red cluster. This result shows the advantage
of learning the linear transformation in addition to the spectral
embedding.

Figure 4(e) and 4(f) show the linear transformed data and
the spectral embedded data when CSP is performed using 20
randomly generated pairwise constraints. We can see that CSP
forms a better spectral embedding using a more scattered linear
transformed data, compared with CSP not using constraints.
The linear transformed data align the four clusters on two
roughly parallel directions, with one direction containing two
Gaussian groups. The spectral embedded data now only has one
red point mixed in the blue cluster. These results demonstrate
that the provided constraints could further improve the
clustering result.

B. Controlled Experiment

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Instance</th>
<th>#Dimension</th>
<th>#Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crabs-Sex</td>
<td>200</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>351</td>
<td>34</td>
<td>2</td>
</tr>
<tr>
<td>Balance-scale</td>
<td>625</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Letters-IJ</td>
<td>1502</td>
<td>16</td>
<td>2</td>
</tr>
</tbody>
</table>

1) Datasets: We evaluate our method on four datasets from
the UCI Machine Learning Repository [1]:

- **Crabs-sex**: This data contains the morphological measure-
ments on two groups Leptograpsus Crabs, one group consisting of the female crabs, and the other consisting of the male crabs.

- **Ionosphere**: This data contains two groups of the radar
data collected by a system in Goose Bay, Labrador. The good radar group returns are those showing evidence of some type of structure in the ionosphere, and bad radar group returns are those that do not; their signals pass through the ionosphere.

- **Balance-scale**: This data set contains psychological experimental results of three groups. Each example is classified as having the balance scale tip to the right, tip to the left, or be balanced.
Fig. 4. A toy example showing the original data and the projected data using CSP. Two clusters are shown with red and blue colors. SP forms spectral embedding of the data. CSP learns a linear transformation of the data and forms spectral embedding of the data based on the transformation.

- **Letters-IJ:** The data is a subset of the letter recognition dataset that contains examples from groups of two different English letters, letter I and letter J.

The dataset information is summarized in Table I.

2) **Experiment Setup:** For each data set, we randomly generate different numbers of pairwise constraints, from 20% \( \times \) \( N \) to \( N \) with a 20% \( \times \) \( N \) increment, where \( N \) is the number of instances in the data. For each number of constraints, we randomly select pairs from the data, and label them as ML.
constraints if they have the same label, and CL constraints otherwise. The experiments are repeated 10 runs and results are averaged.

We compare our method CSP with the unsupervised spectral clustering (SP), the constrained spectral clustering method spLearn in [7], the distance metric learning method xing in [14], and the distance learning method in [5].

Clustering results are evaluated using the Purity metric that compares the clustering solution against the ground truth class labels. To evaluate purity, each instance is first labeled with the majority class in its cluster. The purity is then computed as the accuracy of such label assignments against the ground truth labels.

For all the spectral clustering based methods, we set the bandwidth as $\sigma = 2$. We set the parameters in CSP as $s_0 = 0.5$, $\alpha = 10$, and $\lambda = 0.1$. The transformed dimension is set to $q = d$, meaning that $W$ forms a linear transformation in the original feature space.

3) Results and Discussion:: Figure 5 reports the averaged results of all the methods using different number of constraints. From the results, we have the following observations:

- Comparing with SP, CSP can use the added pairwise constraint to improve the clustering, demonstrating the effectiveness of CSP in using pairwise constraints to aid clustering.
- Comparing with the constrained spectral clustering method spLearn, CSP is more stable with the added constraints. Moreover, CSP could achieve better clustering result on all of the datasets. This shows the advantage of learning the distance metric for spectral clustering.
- Comparing with the distance metric learning methods, CSP outperforms xing’s method on all of the datasets, and is comparable with ITML on most of the datasets, which again demonstrates the effectiveness of our method.

The fact that CSP is not overwhelmingly outperform ITML points us some ways to improve our method. The ITML learns a distance metric that is not far from the Euclidean distance. However, we don’t have such objective in our formulation. So we suspect that one direction of improving our method is to also include such objective in our formulation.

VII. CONCLUSION

In this paper, we consider the problem of spectral clustering with pairwise constraints. We propose a novel approach that alternatively learns a distance metric from such constraints,
and finds the clustering solution on the spectral embedding of the data in the learned metric space. Although our formulation is non-convex, the alternative optimization approach is shown to be practically efficient. We evaluated the proposed method on benchmark datasets from machine learning UCI repository. The results show that our method outperforms both existing methods that only learns distance metric, and existing constrained spectral clustering method without distance metric learning.

**Contributions of Individual Team members**

Teresa mainly worked on using CVX to solve the problem. Yuani mainly worked on the alternative optimization approach. We worked together on the proposal, midterm report, poster, and final report.

**References**