Generalized Cluster Aggregation*

Fei Wang, Xin Wang and Tao Li
Department of Computing and Information Sciences
Florida International University, Miami, FL 33199
{feiwang,xwang009,taoli}@cs.fiu.edu

Abstract

Clustering aggregation has emerged as an important extension of the classical clustering problem. It refers to the situation in which a number of different (input) clusterings have been obtained for a particular data set and it is desired to *aggregate* those clustering results to get a better clustering solution. In this paper, we propose a unified framework to solve the clustering aggregation problem, where the aggregated clustering result is obtained by minimizing the (weighted) sum of the Bregman divergence between it and all the input clusterings. Moreover, under our algorithm framework, we also propose a novel cluster aggregation problem where some must-link and cannot-link constraints are given in addition to the input clusterings. Finally the experimental results on some real world data sets are presented to show the effectiveness of our method.

1 Introduction

Aggregation/Ensemble methods, such as bagging [Breiman, 1996] and boosting, have been widely used in supervised classification to make the results more stable and robust. In fact, data clustering usually suffers from the stability/robustness problems as well because (1) the off-the-shelf clustering methods may discover very different structures in a given set of data because of their different objectives; (2) for a single clustering algorithm, there is no ground truth against which the clustering result can be validated so no cross validation can be performed to tune the parameters; (3) some iterative methods (such as *k-means*) are highly dependent on its initialization. To improve the quality of the clustering results, the idea of aggregation has also been brought into the field of clustering. The problem of cluster aggregation can be described as: *how to make use of a set of different (input) clusterings that have been obtained for a particular data set to find a single final (consensus) clustering that is better than existing clusterings in some sense*.

*The work is partially supported by NSF grants IIS-0546280, DMS-0844513 and CCF-0830659.

During the last decade, many algorithms have been proposed to solve the clustering aggregation problem, e.g., the graph cut method [Fern and Brodley, 2004], information-theoretic method [Topchy *et al.*, 2003][Strehl *et al.*, 2002], matrix factorization based method [Li *et al.*, 2007], and the Bayesian method [Wang *et al.*, 2009]. Most of the traditional approaches treat each input clustering equally. Recently, some researchers proposed to weigh different clusterings differently when performing cluster aggregation to further improve the diversity and reduce the redundancy in combining the input clusterings [Al-Razgan and Domeniconi, 2006][Li and Ding, 2008][Fern and Lin, 2008]. Now cluster aggregation has been widely applied in many AI areas such as computer vision [Yu *et al.*, 2008], information retrieval [Sevillano *et al.*, 2006] and bioinformatics [Hu and Yoo, 2004].

In this paper, we propose a novel cluster aggregation method based on the *cluster connectivity matrix* (*CCM*) [Li *et al.*, 2007], where we aim to find an optimal clustering of the data set whose CCM is *consensus* with respect to the CCMs of the input clusterings. We use *Bregman divergence* [Banerjee *et al.*, 2004] to measure the quantity of such consensus and the resulting problem is a convex one which can be efficiently solved. We also formulate the weighted cluster aggregation problem within our framework and derive a block coordinate descent algorithm to solve it. Moreover, we also show that prior knowledge on the cluster structures, such as pairwise constraints which indicate whether a pair of data points belong to the same cluster, can be easily incorporated into the framework. We also derive a novel approach, *semi-supervised cluster aggregation*, to utilize these constraints in cluster aggregation. Finally the experimental results are presented to show the effectiveness of our method.

It is worthwhile to highlight several aspects of our method:

- Unlike some traditional methods which use Euclidean distance or KL divergence to measure matrix consensus, we adopt a much general criterion – Bregman divergence, which can include many commonly used distortion measures as its special cases.
- We extend our framework to formulate the weighted cluster aggregation problem. We show that both the unweighted and weighted problems are convex and can be efficiently solved.
- We also generalize our framework to incorporate pair-

Table 1: Some distances and the corresponding optimal $\mathbf{M}$

	$\mathcal{D}(x)$	$\phi(x)$	$\nabla\phi(x)$	$\mathbf{M}_{pq}$
Euclidean Distance	$\mathbb{R}$	$\frac{1}{2}x^2$	x	$\frac{1}{n}\sum_i \mathbf{M}^i_{pq}$
Exponential Distance	$\mathbb{R}$	$\exp(x)$	$\exp(x)$	$\log\left(\frac{1}{n}\sum_i \exp(\mathbf{M}^i_{ij})\right)$
Kullback-Leibler Divergence	$\mathbb{R}_{++}$	$x\log x - x$	$\log x$	$\prod_i \exp\left(\frac{1}{n}\log \mathbf{M}^i_{pq}\right)$
Itakura-Saito Distance	$\mathbb{R}_{++}$	$-\log x$	$\frac{1}{x}$	$n/\left(\sum_i \frac{1}{\mathbf{M}^i_{pq}}\right)$
Logistic Distance	$[0,1]$	$x\log x + (1-x)\log(1-x)$	$\log\frac{x}{1-x}$	$\frac{\exp(m_{pq})}{\exp(m_{pq})+1}, m_{pq} = \frac{1}{n}\sum_i \log\frac{\mathbf{M}^i_{pq}}{1-\mathbf{M}^i_{pq}}$

wise constraints, which have widely been used in semi-supervised clustering [Basu *et al.*, 2008] but rarely in cluster aggregation, and we show that the resulting problem can be efficiently solved.

2 Unsupervised Cluster Aggregation with Bregman Divergence

Before we go into the details of our framework, some frequently used notations will be introduced. Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ be a set of n data points. Suppose we are given a set of m clusterings (or partitions) $\mathcal{P} = \{\mathcal{P}^1, \mathcal{P}^2, \cdots, \mathcal{P}^m\}$ of the data in $\mathcal{X}$, each partition $\mathcal{P}^i$ $(i = 1, 2, \cdots, m)$ consists of a set of clusters $\{\pi^i_1, \pi^i_2, \cdots, \pi^i_k\}$, where k is the number of clusters for partition $\mathcal{P}^i$ and $\mathcal{X} = \bigcup_{j=1}^k \pi^i_j$. Note that the number of clusters k could be different for different clusterings.

2.1 Generalized Cluster Aggregation

We define the *connectivity matrix* $\mathbf{M}^i$ for partition $\mathcal{P}^i$ as

$$\mathbf{M}^i_{uv} = \begin{cases} 1, & \text{if } \mathbf{x}_u \text{ and } \mathbf{x}_v \text{ belong to the same cluster} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Thus $\mathbf{M}^i$ is a n by n symmetric square matrix which can be used to represent partition $\mathcal{P}^i$ (in some cases we can also get a *soft* connectivity matrix $\mathbf{M}^i$ such that $\mathbf{M}^i_{uv}$ denotes the possibility that $\mathbf{x}_u$ and $\mathbf{x}_v$ belong to the same cluster, in this case $\mathbf{M}^i_{uv} \in [0, 1]$). Then a general way for finding a *consensus* partition $\mathcal{P}^*$ is to minimize

$$\mathcal{J}_1 = \sum\nolimits_{i=1}^{m} D_\phi(\mathbf{M}, \mathbf{M}^i) \tag{2}$$

where D_ϕ denotes any *separable Bregman divergence* as

$$D_\phi(\mathbf{M}, \mathbf{M}^i) = \sum\nolimits_{u,v} D_\phi\left(\mathbf{M}_{uv}, \mathbf{M}^i_{uv}\right) \tag{3}$$

and

$$D_\phi(x, y) \triangleq \phi(x) - \phi(y) - \nabla\phi(y)(x - y) \tag{4}$$

where $\phi : \mathcal{S} \subseteq \mathbb{R} \to \mathbb{R}$ is a strictly convex function. Therefore $\nabla_{\mathbf{M}}\mathcal{J}_1$ can be computed as:

$$\frac{\partial}{\partial \mathbf{M}_{pq}} \sum_i \sum_{uv} \phi(\mathbf{M}_{uv}) - \phi\left(\mathbf{M}^i_{uv}\right) - \nabla\phi\left(\mathbf{M}^i_{uv}\right)\left(\mathbf{M}_{uv} - \mathbf{M}^i_{uv}\right)$$
$$= \sum_i \nabla\phi\left(\mathbf{M}_{pq}\right) - \nabla\phi\left(\mathbf{M}^i_{pq}\right) = n\nabla\phi\left(\mathbf{M}_{pq}\right) - \sum_i \nabla\phi\left(\mathbf{M}^i_{pq}\right)$$

Let $\nabla_{\mathbf{M}}\mathcal{J}_1 = 0$ then we can get

$$\nabla\phi\left(\mathbf{M}_{pq}\right) = \frac{1}{n}\sum_i \nabla\phi\left(\mathbf{M}^i_{pq}\right) \tag{5}$$

Since $\mathcal{J}_1$ is convex in $\mathbf{M}$, the solution to Eq.(5) is the global optimum for minimizing $\mathcal{J}_1$. Some typical Bregman divergence measures and their corresponding connectivity matrices are summarized in Table 1.

2.2 Generalized Weighted Cluster Aggregation

The objective in Eq.(2) treats each partition equally. However, in real world cases we may want to treat different partition with different importance. Therefore we propose to introduce a set of weighting factors $\{w_i\}_{i=1}^m$ subject to $\forall\ i = 1, 2, \cdots, m,\ w_i \geq 0,\ \sum_i w_i = 1$ and minimize the following weighted objective

$$\mathcal{J}_2 = \sum\nolimits_i w_i D_\phi\left(\mathbf{M}, \mathbf{M}^i\right) \tag{6}$$

The problem is not convex with respect in $\mathbf{w} = (w_1, w_2, \cdots, w_m)^T$ and $\mathbf{M}$. We propose a block coordinate descent algorithm above. Keeping one variable fixed, the optimization over the other is a convex problem with a unique solution. This guarantees monotonic decrease of the objective function and convergence to a stationary point.

Fixing $\mathbf{w}$, the resulted problem is similar to the problem of minimizing $\mathcal{J}_1$, and we can get the solution by setting

$$\nabla_{\mathbf{M}}\mathcal{J}_2 = 0 \tag{7}$$

We can get that

$$\nabla\phi\left(\mathbf{M}_{pq}\right) = \sum\nolimits_i w_i \nabla\phi\left(\mathbf{M}^i_{pq}\right) \tag{8}$$

Fixing $\mathbf{M}$, the problem becomes a linear programming problem

$$\begin{aligned} \min\nolimits_{\mathbf{w}} \quad & \sum\nolimits_i w_i D_\phi\left(\mathbf{M}, \mathbf{M}^i\right) \\ s.t. \quad & w_i \geq 0,\ \sum\nolimits_i w_i = 1 \end{aligned} \tag{9}$$

which is a *linear programming* problem and can be efficiently solved. However, the solution will always be

$$w_i = \begin{cases} 1, & \text{if } i = arg\min_j D_\phi(\mathbf{M}, \mathbf{M}^j) \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

To avoid such trivial solution, we propose to add one regularization term on $\mathcal{J}_2$ and solve the following optimization

problem

$$\begin{aligned} \min_{\mathbf{M},\mathbf{w}} \quad & \sum_{i=1}^{m} w_i D_\phi(\mathbf{M}, \mathbf{M}^i) + \lambda \|\mathbf{w}\|^2 \\ s.t. \quad & \sum_{i=1}^{m} w_i = 1,\ w_i \geq 0\ (\forall\ i = 1, 2, \cdots, m) \end{aligned} \tag{11}$$

where $\lambda > 0$ is a tradeoff parameter. In this way, we will solve a *quadratic programming* problem with respect to $\mathbf{w}$ when fixing $\mathbf{M}$.

Finally, the generalized weighted cluster aggregation (generalized WCA) algorithm is summarized in Algorithm 1.

Algorithm 1 GENERALIZED WCA

Require: Partitions $\{\mathcal{P}^i\}_{i=1}^m$, precision ϵ

1: Construct m cluster aggregation matrix $\{\mathbf{M}^i\}_{i=1}^m$
2: Initialize $\mathbf{w}^0 = [1/m, 1/m, \cdots, 1/m]^T \in \mathbb{R}^{m\times 1}$, $\mathbf{M}^0 = \mathbf{I}_{n\times n}$, $\Delta = +\infty$, $t = 0$
3: **while** $\Delta > \epsilon$ **do**
4: $t = t + 1$
5: Solve $\mathbf{M}^t$ by minimizing $\sum_i w_i D_\phi(\mathbf{M}^t, \mathbf{M}^i)$
6: Solve $\mathbf{w}^t$ by

$$\begin{aligned} \min_{\mathbf{w}^t} \quad & \sum_{i=1}^{m} w_i^t D_\phi(\mathbf{M}^t, \mathbf{M}^i) + \lambda \|\mathbf{w}^t\|^2 \\ s.t. \quad & \sum_{i=1}^{m} w_i^t = 1,\ w_i^t \geq 0\ (\forall\ i = 1, 2, \cdots, m) \end{aligned}$$

7: Compute $\Delta = \|\mathbf{M}^t - \mathbf{M}^{t-1}\|_F$
8: **end while**

3 Semi-supervised Cluster Aggregation with Bregman Divergence

In this section we will consider a novel cluster aggregation setting: in addition to the m partitions, we are also given two sets of pairwise constraints $\mathcal{M}$ and $\mathcal{C}$, such that $(\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{M}$ indicates that $\mathbf{x}_p$ and $\mathbf{x}_q$ belong to the same cluster; and $(\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{C}$ denotes they belong to different clusters. Such constraints has widely been used in *semi-supervised clustering* [Basu *et al.*, 2004][Basu *et al.*, 2008][Wang *et al.*, 2008], however, to the best of our knowledge, there are rarely any works on applying those constraints into cluster aggregation.

3.1 Generalized Cluster Aggregation with Constraints

To incorporate the constraints in $\mathcal{M}$ and $\mathcal{C}$ into the process of generalized cluster aggregation, we need to solve the following problem:

$$\begin{aligned} \min_{\mathbf{M}} \quad & \mathcal{J}_1 = \sum\nolimits_i D_\phi(\mathbf{M}, \mathbf{M}^i) \\ s.t. \quad & \mathbf{M}_{pq} = 1,\ \text{if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{M} \\ & \mathbf{M}_{pq} = 0,\ \text{if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{C} \end{aligned} \tag{12}$$

Clearly, problem (12) is a convex optimization problem with linear constraints. We first rewrite it as

$$\begin{aligned} \min_{\mathbf{M}} \quad & \mathcal{J}_1 = \sum\nolimits_i D_\phi(\mathbf{M}, \mathbf{M}^i) \\ s.t. \quad & (\mathbf{e}_{k_p})^T \mathbf{M} \mathbf{e}_{k_q} = b_k,\ k = 1, 2, \cdots, K \end{aligned} \tag{13}$$

where $\mathbf{e}_{k_i} \in \mathbb{R}^{n\times 1}$ is a indicator vector with only the $k_i i$-th element being one and all other elements being zero, and k is the index of the constraint and K is the total number of constraints. $b_k = 0$ if $(\mathbf{x}_{k_p}, \mathbf{x}_{k_q}) \in \mathcal{C}$ and $b_k = 1$ if $(\mathbf{x}_{k_p}, \mathbf{x}_{k_q}) \in \mathcal{M}$. Now we introduce a set of Lagrangian multipliers $\{\alpha_i\}_{i=1}^k$ and construct the Lagrangian for problem (13) as

$$\mathcal{L} = \sum\nolimits_i D_\phi(\mathbf{M}, \mathbf{M}^i) + \sum\nolimits_k \alpha_k \left((\mathbf{e}_{k_p})^T \mathbf{M} \mathbf{e}_{k_q} - b_k \right) \tag{14}$$

Then

$$\frac{\partial \mathcal{L}}{\partial \mathbf{M}} = n\nabla\phi(\mathbf{M}) - \sum_i \nabla\phi(\mathbf{M}^i) + \sum_k \alpha_k \mathbf{e}_{k_p} \mathbf{e}_{k_q}^T \tag{15}$$

$$\frac{\partial \mathcal{L}}{\partial \alpha_k} = (\mathbf{e}_{k_p})^T \mathbf{M} \mathbf{e}_{k_q} - b_k \tag{16}$$

Let $\partial\mathcal{L}/\partial\mathbf{M} = 0$ and $\partial\mathcal{L}/\partial\alpha_k = 0$ then we can get the following equations

$$\nabla\phi(\mathbf{M}) = \frac{1}{n}\sum_i \nabla\phi(\mathbf{M}^i) + \sum_k \frac{\alpha_k}{n} \mathbf{e}_{k_p} \mathbf{e}_{k_q}^T \tag{17}$$

$$(\mathbf{e}_{k_p})^T \mathbf{M} \mathbf{e}_{k_q} = b_k \tag{18}$$

At the first glance, this is a differential equation group which is hard to solve, however, as we only consider separable Bregman divergence in this paper, and $(\mathbf{e}_{k_p})^T \mathbf{M} \mathbf{e}_{k_q} = \mathbf{M}_{k_p k_q}$, then we can derive the solution of $\mathbf{M}$ as

- For *regular* elements

$$\nabla\phi(\mathbf{M}_{pq}) = \frac{1}{n}\sum_i \nabla\phi(\mathbf{M}_{pq}^i) \tag{19}$$

- For *constrained* elements

$$\alpha_k = n\nabla\phi(\mathbf{M}_{k_p k_q}) - \sum_i \nabla\phi(\mathbf{M}_{k_p k_q}^i) \tag{20}$$

$$\mathbf{M}_{k_p k_q} = b_k \tag{21}$$

where we call $\mathbf{M}_{pq}$ as a regular element if $(\mathbf{x}_p, \mathbf{x}_q) \notin \mathcal{M}$ and $(\mathbf{x}_p, \mathbf{x}_q) \notin \mathcal{C}$; we call $\mathbf{M}_{pq}$ as a constrained element if either $(\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{M}$ or $(\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{C}$. Comparing with Eq.(5), we can find that the solutions for regular elements in $\mathbf{M}$ between semi-supervised and unsupervised generalized cluster aggregation are the same; for constrained elements in $\mathbf{M}$, according to Eq.(21), the semi-supervised algorithm just set them to the exact values in their constraints.

3.2 Generalized Weighted Cluster Aggregation with Constraints

In the semi-supervised weighted cluster aggregation setting, we aim to solve the following problem

$$\begin{aligned} \min_{\mathbf{w},\mathbf{M}} \quad & \sum\nolimits_i w_i D_\phi(\mathbf{M}, \mathbf{M}^i) + \lambda \|\mathbf{w}\|^2 \\ s.t. \quad & \mathbf{M}_{pq} = 1,\ \text{if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{M} \\ & \mathbf{M}_{pq} = 0,\ \text{if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{C} \\ & \sum\nolimits_{i=1}^{m} w_i = 1,\ w_i \geq 0\ (\forall i = 1, \cdots, m) \end{aligned} \tag{22}$$

The same as problem (11), although it is not obvious whether problem (22) with respect to $\mathbf{w}$ and $\mathbf{M}$ jointly, it is convex with respect to them separately with the other one fixed. Therefore we can adopt the block coordinate descent approach to solve the problem. Specifically, when $\mathbf{w}$ is fixed, we should solve the optimal $\mathbf{M}$ by

$$\begin{aligned} \min_{\mathbf{M}} \quad & \sum\nolimits_i w_i D_\phi(\mathbf{M}, \mathbf{M}^i) \\ s.t. \quad & \mathbf{M}_{pq} = 1, \text{ if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{M} \\ & \mathbf{M}_{pq} = 0, \text{ if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{C} \end{aligned} \tag{23}$$

Similar to solving problem (12), we can get the solution to the above problem as

- If $\mathbf{M}_{pq}$ is a constrained element, then

$$\mathbf{M}_{pq} = \begin{cases} 1, & \text{if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{M} \\ 0, & \text{if } (\mathbf{x}_p, \mathbf{x}_q) \in \mathcal{C} \end{cases} \tag{24}$$

- If $\mathbf{M}_{pq}$ is a regular element, then $\mathbf{M}_{pq}$ can be solved by

$$\nabla\phi\left(\mathbf{M}_{pq}\right) = \sum\nolimits_i w_i \nabla\phi\left(\mathbf{M}^i_{pq}\right) \tag{25}$$

If $\mathbf{M}$ is fixed, then we can solve the optimal $\mathbf{w}$ by

$$\begin{aligned} \min_{\mathbf{w}} \quad & \sum\nolimits_{i=1}^{m} w_i D_\phi(\mathbf{M}^t, \mathbf{M}^i) + \lambda \|\mathbf{w}\|^2 \\ s.t. \quad & \sum\nolimits_{i=1}^{m} w_i = 1,\ w_i \geq 0\ (\forall\, i = 1, \cdots, m) \end{aligned} \tag{26}$$

which is a convex QP problem and can be efficiently solved.

Algorithm 2 summarizes the basic procedure of generalized weighted cluster aggregation (WCA) with constraints.

Algorithm 2 GENERALIZED WCA WITH CONSTRAINTS

Require: Partitions $\{\mathcal{P}^i\}_{i=1}^m$, constraint sets $\mathcal{M}$ and $\mathcal{C}$, precision ϵ

1: Construct m cluster aggregation matrix $\{\mathbf{M}^i\}_{i=1}^m$
2: Initialize $\mathbf{w}^0 = [1/m, 1/m, \cdots, 1/m]^T \in \mathbb{R}^{m\times 1}$, $\mathbf{M}^0 = \mathbf{I}_{n\times n}$, $\Delta = +\infty$, $t = 0$
3: **while** $\Delta > \epsilon$ **do**
4: $\quad t = t + 1$
5: $\quad$ Obtain $\mathbf{M}^t$ by solving problem (23) with $\mathbf{w} = \mathbf{w}^t$
6: $\quad$ Obtain $\mathbf{w}^t$ by solving problem (26) with $\mathbf{M} = \mathbf{M}^t$
7: $\quad$ Compute $\Delta = \|\mathbf{M}^t - \mathbf{M}^{t-1}\|_F$
8: **end while**

4 Experiments

In this section we will present a set of experiments to test the effectiveness of the proposed *generalized cluster aggregation* (*GCA*) method. The data sets used in our experiments include both synthetic and real world data sets.

4.1 Toy Examples

In this section we construct two synthetic data sets to test the power of the proposed GCA method. Specifically, the two data sets are constructed in the following way:

- **Clusters with different shapes**. This data set contains five clusters with shapes like letter I (100 points), J (200 points), C (183 points), A (223 points), I (100 points). The distribution of the data set is shown in Fig.1(a).
- **Clusters lie in different subspaces**. Points with four clusters, each of which exists in just two dimensions with the third dimension being noise [Parsons *et al.*, 2004]. The first two clusters exist in dimensions x and y. The data forms a normal distribution with means 0.6 and -0.6 in dimension x and 0.5 in dimension y, and standard deviations of 0.1. In dimension z, these clusters have $\mu = 0$ and $\sigma = 1$. The second two clusters are in dimensions y and z and are generated in the same manner.

We first run K-means on these data sets with randomly initialized cluster centers, and the number of clusters is set to the number of true clusters. The results are shown in Fig.1(b)(e), from which we can see that K-means is totally confused in these cases. We also test the performance of our *generalized weighted cluster aggregation* (*GWCA*) algorithm (see section 2.2) under the Euclidean distance, where the 30 base cluster connectivity matrices are obtained by clustering the data set into 20 clusters using K-means with randomly initialized cluster centers. The weight vector is initialized to $\mathbf{w} = [1/30, 1/30, \cdots, 1/30]^T \in \mathbb{R}^{30\times 1}$. Finally the clustering result is obtained by running K-means on the aggregated cluster connectivity matrix, and the results are shown in Fig.1(c)(f), which clearly illustrate the superiority of our method.

4.2 Experiments on Real World Data Sets

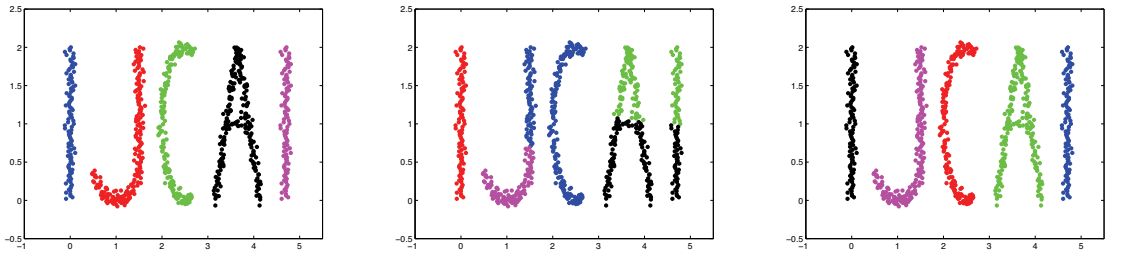
In this section we will present the results of applying our algorithm to a set of real world data sets. First we will describe the basic characteristics of the data sets.

Data Sets

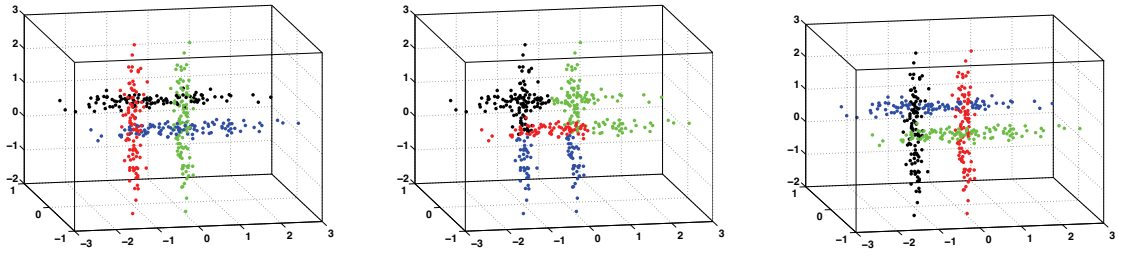
We use totally 14 data sets to evaluate the effectiveness of our proposed. The basic information of these data sets are summarized in Table 2. In the following we will briefly introduce these data sets.

• **UCI Data Sets**. These data sets are from the UCI Repository, which include **Digits 389**, **Glass**, **Ionosphere**, **Iris**, **Letter IJL**, **Protein**, **Soybean**, **Wine**, and **Zoo**, where **Digits 389** is a randomly sampled subset of the handwritten digits 3, 8, 9 from the **digits** data set, and **Letter IJL** is a randomly sampled subset containing I, J, L from the **letters** data set.

• **Text Data Sets**. These data sets are standard benchmark data sets that commonly used to assess text clustering algorithms. The **CSTR** data set contains 476 abstracts of technical reports published in the Department of Computer Science at a research university; The **Log** data set contains 1367 text messages of system log from different desktop machines describing the status of computer components; The **Reuters** data set is a subset of the Reuters-21578 Text Categorization Test collection contains 10 most frequent categories among the 135 topics; The **WebACE** data set was from WebACE project, which contains 2340 documents consisting news articles from Reuters new service via the Web in October 1997[Han *et al.*, 1998]; The **WebKB** data set contains web pages gathered



(a) Original Data Set (b) K-means Result (c) GWCA with Euclidean Distance



(d) Original Data Set (e) K-means Result (f) GWCA with Euclidean Distance

Figure 1: Toy examples. (a),(d) are the original data sets, where we use different colors to denote different clusters. (b),(e) are the clustering results by K-means. (c),(f) are the results of our weighted clustering aggregation method with Euclidean distance.

Table 2: Description of the Data Sets

Data Sets	# Samples	# Dimensions	# Class
CSTR	476	1000	4
Digits 389	456	16	3
Glass	214	9	7
Ionoshpere	351	34	2
Iris	150	4	3
Protein	116	20	6
Letter IJL	227	16	3
Log	1367	200	8
Reuters	2900	1000	10
Soybean	47	35	4
WebACE	2340	1000	20
WebKB4	4199	1000	4
Wine	178	13	3
Zoo	101	18	7

from university computer science departments. We use a subset containing categories student, faculty, course and project. All the data sets are preprocessed by the rainbow package.

Evaluation Measure

We use *Clustering Accuracy* to measure the performance of our proposed methods. It discovers the one-to-one relationship between clusters and classes and measures the extent to which each cluster contained data points from the corresponding class. It sums up the whole matching degree between all pair class-clusters. Clustering accuracy can be computed as:

$$Acc = \frac{1}{N} \max \left(\sum_{\mathcal{C}_k, \mathcal{L}_m} T(\mathcal{C}_k, \mathcal{L}_m) \right), \qquad (27)$$

where $\mathcal{C}_k$ denotes the k-th cluster in the final results, and $\mathcal{L}_m$ is the true m-th class. $T(\mathcal{C}_k, \mathcal{L}_m)$ is the number of entities which belong to class m are assigned to cluster k. Accuracy computes the maximum sum of $T(\mathcal{C}_k, \mathcal{L}_m)$ for all pairs of clusters and classes, and these pairs have no overlaps. Greater clustering accuracy means the better clustering performance.

Comparative Methods

Besides our method, we also show the experimental results of some other methods for comparison including

- **K-means**, which is randomly initialized and the results are averaged over 50 independent runs.
- **Spectral Clustering** (**SC**), which is implemented in the same way as in [Yu and Shi, 2003].
- **Cluster-based Similarity Partitioning Algorithm** (**CSPA**) and **Hyper-Graph-Partitioning Algorithm** (**HGPA**), implemented as in [Strehl *et al.*, 2002].
- **Nonnegative Matrix Factorization based Consensus clustering** (**NMFC**), implemented as in [Li *et al.*, 2007].
- **Weighted Consensus clustering** (**WC**). The implementation is the same as in [Li and Ding, 2008].

We present the results of our GWCA algorithm and the Semi-Supervised GWCA (SSGWCA) algorithm under the Euclidean and exponential distance (denoted as EGWCA and eGWCA). For the semi-supervised approaches, we first randomly label 10% of the data, and then use these data labels to generate the constraint sets $\mathcal{M}$ and $\mathcal{C}$.

Experimental Results

The experimental results are summarized in Table 3, where for our GWCA series methods, we run spectral clustering on

Table 3: Experimental Results in Clustering Accuracy

	K-means	SC	CSPA	HPGA	NMFC	WC	EGWCA	eGWCA	SSEGWCA	SSeGWCA
CSTR	0.45	0.54	0.50	0.62	0.56	0.64	0.64	0.67	0.75	0.78
Digits 389	0.59	0.69	0.78	0.38	0.73	0.71	0.72	0.70	0.78	0.76
Glass	0.38	0.44	0.43	0.40	0.49	0.49	0.50	0.52	0.59	0.60
Ionosphere	0.70	0.74	0.68	0.52	0.71	0.71	0.73	0.72	0.77	0.78
Iris	0.83	0.91	0.86	0.69	0.89	0.89	0.92	0.90	0.97	0.96
Protein	0.53	0.58	0.59	0.59	0.60	0.63	0.65	0.67	0.75	0.77
Log	0.61	0.58	0.47	0.43	0.71	0.69	0.73	0.72	0.80	0.79
LetterIJL	0.49	0.48	0.48	0.53	0.52	0.52	0.54	0.56	0.67	0.65
Reuters	0.45	0.43	0.43	0.44	0.43	0.44	0.46	0.44	0.55	0.57
Soybean	0.72	0.77	0.70	0.81	0.89	0.91	0.90	0.90	0.96	0.97
WebACE	0.41	0.39	0.40	0.42	0.48	0.46	0.47	0.49	0.58	0.60
WebKB4	0.60	0.58	0.61	0.62	0.64	0.63	0.65	0.67	0.76	0.75
Wine	0.68	0.68	0.69	0.52	0.70	0.72	0.73	0.73	0.80	0.82
Zoo	0.61	0.64	0.56	0.58	0.62	0.70	0.72	0.75	0.84	0.87

the final combined cluster aggregation matrix (i.e., which is used as the similarity matrix). From Table 3 we can clearly observe that our GWCA based algorithm can generate better clusterings, and the results can be further improved with pairwise instance-level constraints.

5 Conclusions

In this paper we propose a general framework for cluster aggregation based on Bregman divergence, and we derive a series of clustering aggregation algorithms under such a framework. Finally the experiments on several real world data sets are presented to show the effectiveness of our method.

References

[Al-Razgan and Domeniconi, 2006] M. Al-Razgan and C. Domeniconi. Weighted clustering ensembles. In *Proceedings of The 6th SIAM International Conference on Data Mining*, 2006.

[Banerjee *et al.*, 2004] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh. Clustering with bregman divergences. In *Proc. of the 4th SIAM International Conference on Data Mining*, 2004.

[Basu *et al.*, 2004] S. Basu, B. Mikhail, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *ACM SIGKDD Conference*, pages 59–68, 2004.

[Basu *et al.*, 2008] S. Basu, I. Davidson, and K. L. Wagstaff, editors. *Constrained Clustering: Advances in Algorithms, Theory, and Applications*. CRC Press, 2008.

[Breiman, 1996] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[Fern and Brodley, 2004] X. Z. Fern and C. E. Brodley. Solving cluster ensemble problems by bipartite graph partitioning. In *Proc. of the 21st Int'l Conference on Machine learning*, 2004.

[Fern and Lin, 2008] X. Z. Fern and W. Lin. Cluster ensemble selection. In *Proceedings of the 8th SIAM International Conference on Data Mining*, pages 787–797, 2008.

[Han *et al.*, 1998] E. H. Han, D. Boley, M. Gini, R. Gross, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore. Webace: a web agent for document categorization and exploration. In *AGENTS '98: Proceedings of the second international conference on Autonomous agents*, pages 408–415, 1998.

[Hu and Yoo, 2004] X. Hu and I. Yoo. Cluster ensemble and its applications in gene expression analysis. In *Proc. of the 2nd conference on Asia-Pacific bioinformatics*, pages 297–302, 2004.

[Li and Ding, 2008] T. Li and C. Ding. Weighted consensus clustering. In *Proceedings of The 8th SIAM International Conference on Data Mining*, pages 798–809, 2008.

[Li *et al.*, 2007] T. Li, C. Ding, and M. I. Jordan. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *Proceedings of the 7th IEEE International Conference on Data Mining*, pages 577–582, 2007.

[Parsons *et al.*, 2004] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: A review. *SIGKDD Explorations, Newsletter of the ACM Special Interest Group on Knowledge Discovery and Data Mining*, 6(1), 2004.

[Sevillano *et al.*, 2006] X. Sevillano, G. Cobo, F. A., and J. C. Socoró. Feature diversity in cluster ensembles for robust document clustering. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 697–698, 2006.

[Strehl *et al.*, 2002] A. Strehl, J. Ghosh, and C. Cardie. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Jour. of Mach. Learn. Res.*, 3:583–617, 2002.

[Topchy *et al.*, 2003] A. Topchy, A. K. Jain, and W. Punch. Combining multiple weak clusterings. pages 331–338, 2003.

[Wang *et al.*, 2008] F. Wang, T. Li, and C. Zhang. Semi-supervised clustering via matrix factorization. In *The 8th SIAM Conference on Data Mining (SDM)*, 2008.

[Wang *et al.*, 2009] H. Wang, H. Shan, and A. Banerjee. Bayesian cluster ensembles. In *Proceedings of the 9th SIAM International Conference on Data Mining*, 2009.

[Yu and Shi, 2003] Stella X. Yu and Jianbo Shi. Multiclass spectral clustering. In *Proceedings of International Conference on Computer Vision*, pages 313–319, 2003.

[Yu *et al.*, 2008] Z. Yu, Z. Deng, H.-S. Wong, and X. Wang. Fuzzy cluster ensemble and its application on 3d head model classification. In *Proceedings of the IEEE IJCNN*, pages 569–576, 2008.