The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Balanced Clustering via Exclusive Lasso: A Pragmatic Approach

Zhihui Li,[1] Feiping Nie,[2*] Xiaojun Chang,[3] Zhigang Ma,[3] Yi Yang[4]
[1]Beijing Etrol Technologies Co., Ltd.
[2]Centre for OPTical Imagery Analysis and Learning, Northwestern Polytechnical University.
[3]School of Computer Science, Carnegie Mellon University.
[4]Centre for Artificial Intelligence, University of Technology Sydney.

Abstract

Clustering is an effective technique in data mining to generate groups that are the matter of interest. Among various clustering approaches, the family of k-means algorithms and min-cut algorithms gain most popularity due to their simplicity and efficacy. The classical k-means algorithm partitions a number of data points into several subsets by iteratively updating the clustering centers and the associated data points. By contrast, a weighted undirected graph is constructed in min-cut algorithms which partition the vertices of the graph into two sets. However, existing clustering algorithms tend to cluster minority of data points into a subset, which shall be avoided when the target dataset is balanced. To achieve more accurate clustering for balanced dataset, we propose to leverage exclusive lasso on k-means and min-cut to regulate the balance degree of the clustering results. By optimizing our objective functions that build atop the exclusive lasso, we can make the clustering result as much balanced as possible. Extensive experiments on several large-scale datasets validate the advantage of the proposed algorithms compared to the state-of-the-art clustering algorithms.

Introduction

Clustering is a fundamental research topic in data mining and is widely used for many applications in the field of artificial intelligence, statistics and social sciences (Jain, Murty, and Flynn 1999; Jain and Dubes 1988a; Girolami 2002; Wu et al. 2012b; Wang, She, and Cao 2013; Ye, Zhao, and Liu 2007a; Nie, Cai, and Li 2017). The objective of clustering is to partition the original data points into a number of groups so that the data points within the same cluster are close to each other while those in different clusters are far away from each other (Jain and Dubes 1988b) (Nie, Xu, and Li 2012) (Chang et al. 2015) (Filippone et al. 2008) (Li et al. 2017).

Among various approaches for clustering, K-means and min-cut are two most popular choices in reality because of their simplicity and effectiveness (Wu et al. 2012a). The general procedure of traditional K-means (TKM) is to randomly initialize c clustering centers, assign each data point to its nearest cluster and compute a new clustering center iteratively. Some researchers claim that the curse of dimensionality may deteriorate the performance of TKM (Ding and Li 2007). A straightforward solution of this problem is to project the original dataset to a low-dimensional subspace by dimensionality reduction, for example, PCA, before performing TKM. Discriminative analysis has been shown effective in enhancing clustering performance (Ding and Li 2007) (la Torre, Fernando, and Kanade 2006) (Ye, Zhao, and Liu 2007b). Motivated by this fact, discriminative k-means (DKM) (Ye, Zhao, and Wu 2007) is proposed to incorporate discriminative analysis and clustering into a single framework to formalize the clustering as a trace maximization problem.

By contrast, the min-cut clustering is realized by constructing a weighted undirected graph and then partitioning its vertices into two sets so that the total weight of the set of edges with endpoints in different sets is minimized (Yuan et al. 2014) (Raj and Wiggins 2010). Among several graph clustering methods, min-cut tends to provide more balanced clusters as compared to other graph clustering criterion. As the within-cluster similarity in min-cut method is explicitly maximized, solving the min-cut clustering problem is nontrivial. The main difficulty lies in the constraint on the solution. Thus, to make this problem tractable, researchers proposed to relax the constraint.

Although k-means and min-cut have achieved promising performance in many applications, they have certain limit.Given that the distribution of the data points is balanced, one would expect the clustering result to reflect such balance. That being said, a clustering algorithm shall avoid partitioning a minority of data points into a cluster. Nonetheless, both K-means and min-cut, as well as some other similar clustering algorithms, do not guarantee balanced clustering result. In many real world data mining applications, the data from each cluster are about the same. For example, the male and female populations in the same age range cannot be very different. Therefore, for those data which are equally distributed, it is more reasonable to explicitly guarantee the clustering results balanced.

Motivated by the limit of k-means and min-cut for handling balanced data, we propose to design a balanced clustering algorithm. Specifically, the exclusive lasso proposed by Zhou *et al.* (Zhou, Jin, and Hoi 2010) has been exploited in our approach to fulfill such purpose. The exclusive lasso was

*Corresponding author. Email: feipingnie@gmail.com

originally used for feature selection across multiple tasks. It models the scenario when variables in the same group compete with each other. With exclusive lasso, if one feature in a group is given a large weight, it tends to assign small or even zero weights to the other features in the same group. Suppose that the exclusive lasso is applied on a bunch of data points across multiple categories. In a similar manner, we introduce a competition among different categories for the same data point. If more data points are clustered into one category, other categories would get fewer data points. The exclusive lass, thus in a sense, measures the balance degree of the clustering result. The smaller value the exclusive lasso obtains, the more balanced the clustering result is. With such insight, we formulate our clustering approach based on minimizing the exclusive lasso. In this paper, we particularly incorporate the exclusive lasso into k-means clustering and min-cut clustering, aiming to promote these two mainstream clustering approaches with stronger ability of attaining balanced clusters.

The major contributions of this paper can be summarized as follows:

- We leverage the exclusive lasso to introduce a competition among different categories for the same data point, thus enhancing the balance of the clustering result.
- The exclusive lasso is particularly tailored for k-means and min-cut. Thus, these two mostly used clustering approaches are able to achieve more balanced clustering result.
- The proposed algorithms are non-smooth and difficult to optimize. We propose a new iterative solution to solve the problems.

Related Work

The Traditional k-means

As one of the most efficient clustering algorithms, k-means clustering has been widely applied to real-world applications. The centroids of clusters are utilized to characterize the data. The objective of k-means is to minimize the sum of the squared errors defined by:

$$J_k = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - m_k\|^2, \tag{1}$$

where $X = (x_1, \ldots, x_n)$ denotes the data matrix and $m_k = \sum_{i \in C_k} x_i / n_k$ is the centroid of a cluster C_k of n_k data points.

Previous work (Wang et al. 2012) has shown that H-orthogonal non-negative matrix factorization (NMF) is equivalent to relaxed k-means clustering. Thus, k-means clustering can be reformulated using the clustering indicator as follows:

$$\min_{F,G} \|X - HF^T\|_F^2 \tag{2}$$

$$s.t.\ G_{ik} \in \{0,1\}, \sum_{k=1}^{K} H_{ik} = 1, \forall i = 1, 2, \ldots, n \tag{3}$$

where $X \in \mathbb{R}^{d \times n}$ is the input data matrix with n data represented by d-dimensional features; $F \in \mathbb{R}^{d \times K}$ is the clustering indicator matrix; $H \in \mathbb{R}^{n \times K}$ is the clustering assignment matrix and each row of H satisfies the 1-of-K coding scheme (if a data point x_i is assigned to the k-th cluster then $H_{ik} = 1$ and $H_{ik} = 0$ otherwise). In this paper, given a matrix $X = \{x_{ij}\}$, its i-th row, j-th column are denoted as x^i, x_j, respectively.

In the literature, the classical K-means and its variants have been applied to many data mining applications. For example, Mehrdad *et al.* (Wang et al. 2012) propose a harmony K-means (HKM) algorithm based on harmony search optimization method and applied it to document clustering. HKM can be proved by means of finite Markov chain theory to converge to the global optimum. Zhang *et al.* (Zhang and Xia 2009) propose a new neighborhood density method for selecting initial cluster centers for K-means clustering. Deepak *et al.* (Turaga, Vlachos, and Verscheure 2009) employ quantization schemes to retain the outcome of clustering operations. Although these methods get good performance, they have not considered how to achieve balanced clustering result when the given data points are evenly distributed. By contrast, we aim to develop a balanced k-means clustering algorithm that well addresses this issue.

Min-Cut

The principle of min-cut is rooted in graph theory. It needs a graph based on a weight matrix $W \in \mathbb{R}^{n \times n}$ built from n data points $\{x_1, \ldots, x_n\}$. The min-cut graph clustering objective function can be generalized as:

$$J = \sum_{1 \le p < q \le K} s(C_p, C_q) + s(C_p, C_q) = \sum_{k=1}^{K} s(C_k, \overline{C_k}) \tag{4}$$

where K is the number of clusters, C_k is the k-th cluster (sub-graph in graph G), $\overline{C_k}$ is the complement of a subset C_k in graph G, and for any set A and B

$$s(A, B) = \sum_{i \in A} \sum_{j \in B} W_{ij}, \ \ d_i = \sum_{j} W_{ij}. \tag{5}$$

We denote q_k $(k = 1, \ldots, K)$ as the cluster indicators where the i-th element of q_k is set to 1 if the i-th data point x_i belongs to the k-th cluster, and 0 otherwise. For example, if the data points within each cluster are adjacent,

$$q_k = (0, \ldots, 0, \overbrace{1, \ldots, 1}^{n_k}, 0, \ldots, 0)^T. \tag{6}$$

After simple mathematical deduction, we can find that

$$s(C_k, \overline{C_k}) = \sum_{i \in C_k} \sum_{j \in \overline{C_k}} W_{ij} = q_k^T (D - W) q_k$$

$$\sum_{i \in C_k} d_i = q_k^T D q_k, \ \ s(C_k, C_k) = q_k^T W q_k, \tag{7}$$

where D is a diagonal matrix with the i-th diagonal element as d_i. The objective function of min-cut method can therefore be reformulated as:

$$J = \sum_{k=1}^{K} q_k^T (D - W) q_k \tag{8}$$

Min-Cut clustering has been applied in various applications. Wang *et al.* (Wang and Davidson 2010) propose a flexible and generalized framework for constrained spectral clustering, interpret the algorithm as finding the normalized min-cut of a labeled graph, and apply it to constrained image segmentation. Dynamic graph clustering algorithm, proposed by (Saha and Mitra 2006) can provide strong theoretical quality guarantee on clusters. However, none of the existing work on min-cut is capable of balanced clustering when necessary, which shall be addressed by our newly proposed balanced min-cut algorithm.

Exclusive Lasso

Zhou *et al.* propose the exclusive lasso to model the scenario when variables in the same group compete with each other. They apply it to multi-task feature selection and obtain good performance. The exclusive lasso (Zhou, Jin, and Hoi 2010) is defined as follows:

$$\|\beta\|_e = \sqrt{\sum_{j=1}^{d} (\sum_{k=1}^{m} \|\beta_k^j\|)^2}, \tag{9}$$

where $\|\beta\|_e$ is a regularizer that controls the complexity of the combination weights.

In (Zhou, Jin, and Hoi 2010), the regularizer introduces an l_1-norm to combine the weights for the same category used by different data points and an l_2-norm to combine the weights of different categories. Since l_1-norm tends to achieve a sparse solution, the construction in the exclusive lasso essentially introduces a competition among different categories for the same data points.

In our work, the exclusive lasso is used as a balance constraint. We will prove that the value of exclusive lasso indicates the balance degree of our clustering algorithms.

The proposed algorithm

Balance Constraint

Given F as a cluster indicator matrix, the exclusive lasso of F is written as

$$\|F\|_e = \sqrt{\sum_{j=1}^{c} (\sum_{i=1}^{n} \|f_{ij}\|)^2}. \tag{10}$$

With simple mathematical deduction, the exclusive lasso can be rewritten as:

$$\|F\|_e = Tr(F^T \mathbf{1}\mathbf{1}^T F). \tag{11}$$

From this equation, we can observe that the value of exclusive lasso equals the square-sum of the number of data points in each class. In the following, we prove that the most balanced clustering can be achieved by minimizing the exclusive lasso.

Theorem 1. *Given $n_1 + n_2 + \cdots + n_k = N$ and $n_i|_{i=1}^k \geq 0$, $\sum_{i=1}^k n_i^2$ arrives at its minimum when $n_i = \frac{N}{k}$.*

Proof. According to the Cauchy inequality,

$$\begin{aligned}&(n_1^2 + n_2^2 + \cdots + n_k^2)(b_1^2 + b_2^2 + \cdots + b_k^2)\\ \geq &(a_1 b_1 + a_2 b_2 + \cdots + a_k b_k)^2\end{aligned}$$

Let $b_i|_{i=1}^k = 1$, the equality holds when $n_1 = n_2 = \cdots = n_k$. Hence, we can easily have the conclusion that when $n_i = \frac{N}{k}$, $\sum_{i=1}^k n_i^2$ get its minimal value. □

According to the above theorem, by minimizing the exclusive lasso, each cluster will have $\frac{n}{c}$ data points. The most balanced clustering result is thus obtained. Hence, we use the the exclusive lasso as the balance constraint.

Balanced k-Means

In the setting of clustering, given n data points $\{x_i\}|_{i=1}^n$, we have a data matrix $X = (x_1, \ldots, x_n) \in \mathbb{R}^{d \times n}$. Our goal in balanced k-means is to partition $\{x_i\}|i = 1^n$ into K balanced clusters among different categories.

Noting that the exclusive lasso is capable of introducing competition among different categories, we apply the exclusive lasso to the classical k-means to obtain balanced clusters. The proposed objective function of balanced k-means is formulated as follows:

$$\min_{F \in Ind} \|X - HF^T\|_F^2 + \gamma \|F\|_e \tag{12}$$

By substituting $\|F\|_e$ with (9), the objective function can be rewritten as follows:

$$\min_{F \in Ind} \|X - HF^T\|_F^2 + \gamma Tr(F^T \mathbf{1}\mathbf{1}^T F) \tag{13}$$

where $F \in Ind$ means $F \in \mathbb{R}^{n \times K}$ is an indicator matrix used for clustering; $H \in \mathbb{R}^{d \times K}$ is the clustering assignment matrix; γ is a parameter.

The optimal H and F would minimize the objective function value. Since it is difficult to compute the optimal H and F simultaneously, we present an iterative approach to optimize this algorithm. To be more specific, we can obtain the optimal H by fixing F by a simple linear equation. Similarly, we can get the optimal F by fixing H.

For a fixed F, by setting the derivative of (13) w.r.t H to zero, we obtain

$$H = XF(F^T F)^{-1} \tag{14}$$

Then we fix H, we update F as follows: we update one row of F each time while fixing the other rows of the prediction matrix F. Specifically, the updating of one row is realized by finding the element being 1 that results in the minimum of (13). We iterate the updating of each row until convergence as shown in Algorithm 1.

Computational Analysis: The computation complexity of Algorithm 1 is $O(K)$. Since the indicator matrix F is sparse, this inverse operation is very efficient. When sufficient computational resources are available and parallel computing is implemented, this algorithm can be solved with desired efficiency.

Algorithm 1 Algorithm to solve the objective function of balanced k-means

Input: Data matrix $X \in \mathbb{R}^{d\times n}$
Output: Indicator matrix $F \in \mathbb{R}^{n\times K}$

1: Initialize the indicator matrix F randomly.
2: **repeat**
3: Fixing F, compute H according to $H = XF(F^TF)^{-1}$
4: Fixing H, update F as follows:
Update each row of F while fixing the remaining rows.
5: **until** CONVERGENCE

Return: Indicator matrix F.

Theorem 2. *Algorithm 1 decreases the objective value of Eq.* (13) *in each iteration.*

Proof. In each iteration t of Algorithm 1, according to Step 3, we know that:

$$H_{t+1} = \min_F \|X - HF_t^T\|_F^2 + \gamma Tr(F_t^T\mathbf{1}\mathbf{1}^TF_t) \quad (15)$$

Thus, we have:

$$\begin{aligned}&\|X - H_{t+1}F_t^T\|_F^2 + \gamma Tr(F_t^T\mathbf{1}\mathbf{1}^TF_t)\\ \leq&\|X - H_tF_t^T\|_F^2 + \gamma Tr(F_t^T\mathbf{1}\mathbf{1}^TF_t)\end{aligned} \quad (16)$$

According to step 4, we obtain:

$$\begin{aligned}&\|X - H_tF_{t+1}^T\|_F^2 + \gamma Tr(F_{t+1}\mathbf{1}\mathbf{1}^TF_{t+1})\\ \leq&\|X - H_tF_t^T\|_F^2 + \gamma Tr(F_t\mathbf{1}\mathbf{1}^TF_t)\end{aligned} \quad (17)$$

Adding Eq. (16) and Eq. (17), we arrive at:

$$\begin{aligned}&\|X - H_{t+1}F_{t+1}^T\|_F^2 + \gamma Tr(F_{t+1}\mathbf{1}\mathbf{1}^TF_{t+1})\\ \leq&\|X - H_tF_t^T\|_F^2 + \gamma Tr(F_t\mathbf{1}\mathbf{1}^TF_t)\end{aligned} \quad (18)$$

which proves that the algorithm decreases the objective function value in each iteration. □

According to Theorem 2, we can see that the value of the objective function (13) decrease at each iteration of Algorithm 1. In addition, it is clear that (13) is lower bounded by 0. Therefore, Algorithm 1 is guaranteed to converge.

Balanced Min-Cut

We similarly aim to cluster n data points $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^{d\times n}$ into K clusters. To begin with, we use the Gaussian function to construct a weight matrix A. The weight A_{ij} is defined as:

$$A_{ij} = \begin{cases} \exp(-\frac{\|x_i - x_j\|^2}{\delta^2}), x_i \; and \; x_j \; are \; k \\ nearest \; neighbors. \\ 0, \; otherwise \end{cases} \quad (19)$$

where δ is utilized to control the spread of neighbors. Given the weight matrix A and the cluster indicator matrix F, the objective function of min-cut graph clustering is formulated as follows:

$$\min_{F\in Ind} \mathbf{1}^TA\mathbf{1} - Tr(F^TAF), \quad (20)$$

which is equivalent to the following objective function:

$$\max_{F\in Ind} Tr(F^TAF) \quad (21)$$

We further incorporate the exclusive lasso into min-cut and get the following objective function:

$$\max_{F\in Ind} Tr(F^TAF) - \gamma\|F\|_e \quad (22)$$

In the same manner, we substitute $\|F\|_e$ with (9) and rewrite the objective function as follows:

$$\max_{F\in Ind} Tr(F^TAF) - \gamma Tr(F^T\mathbf{1}\mathbf{1}^TF) \quad (23)$$

With a simple mathematical deduction, the objective function is rewritten as:

$$\max_{F\in Ind} Tr\left(F^T(\rho I + A - \gamma\mathbf{1}\mathbf{1}^T)F\right), \quad (24)$$

where ρ is a large enough constant to make $\rho I + A - \gamma\mathbf{1}\mathbf{1}^T$ positive-definite. Defining $B = (\rho I + A - \gamma\mathbf{1}\mathbf{1}^T)F$, we update F by solving $\max_{F\in Ind} Tr(F^TB)$. F is iteratively updated until convergence as shown in Algorithm 2.

Algorithm 2 Algorithm to solve the objective function of balanced min-cut

Input: Data matrix X
Output: Indicator matrix F

1: Compute the weight matrix A using Eq (19).
2: **repeat**
3: Compute B according to $B = (\rho I + A - \gamma\mathbf{1}\mathbf{1}^T)F$
4: Update F by solving $\max_{F\in Ind} Tr(F^TB)$
5: **until** CONVERGENCE

Return: Indicator matrix F

Theorem 3. *Algorithm 2 increases the objective function value of Eq.* (24) *in each iteration.*

Proof. In the Steps 3 and 4 of Algorithm 2, we denote the updated B and F by $\hat{B}$ and $\hat{F}$, respectively. Since the updated B and F are the optimal solutions of the problem $\max_{F\in Ind} Tr(F^TB)$, we have:

$$Tr(\hat{F}^T(\rho I + A - \gamma\mathbf{1}\mathbf{1}^T)F) \geq Tr(F^T(\rho I + A - \gamma\mathbf{1}\mathbf{1}^T)F), \quad (25)$$

which proves that the algorithm increase the objective function value in each iteration. □

According to Theorem 3, we can observe that the value of objective function (24) increases at each iteration of Algorithm 2. Therefore, Algorithm 2 is proved to converge.

Experiment

In this section, extensive experiments are conducted to evaluate the proposed clustering methods. We give two sets of experiments. The first one is to compare the proposed balanced K-means clustering to K-means based clustering algorithms, including the classical K-means (KM) clustering, DisCluster (DC), DisKmeans (DKM) clustering (Ye, Zhao, and Wu 2007), AKM (Wang et al. 2012) and HKM (Wang et al. 2012). The second one is to compare the proposed balanced min-cut clustering to the classical min-cut clustering, MinMax Cut clustering, Ratio Cut clustering and Normalized Cut clustering algorithms.

Datasets

A variety of datasets are used in our experiments which are described as follows.

– MNIST Handwritten Digit Dataset: The MNIST handwritten digit dataset (LeCun et al. 2011) is a large-scale dataset of handwritten digits. It is widely used as a test bed in data mining. The dataset contains 60,000 training images and 10,000 testing images. We merge all the training and testing images in the experiments. The pixel values are used as feature representation.

– USPS handwritten digit dataset: We additionally use the USPS dataset to validate the performance on handwritten digit recognition. The dataset consists of 9298 gray-scale handwritten digit images. We resize the images to 16×16 and use pixel values as the features.

– YaleB face dataset: The YaleB dataset (Georghiades, Belhumeur, and Kriegman 2001) contains 2414 near frontal images from 38 persons under different illuminations. Each image is resized to 32×32 and the pixel value is used as feature representation.

– ORL face dataset: The ORL dataset (Samaria and Harter 1994) consists of 40 different subjects with 10 images each. We also resize each image to 32×32 and use pixel values to represent the images.

– JAFFE Japanese Female Facial Expression dataset: The JAFFE dataset (Lyons, Budynek, and Akamatsu 1999) consists of 213 images of different facial expressions from 10 different Japanese female models. The images are resized to 26×26 and represented by pixel values.

– HumanEVA Motion dataset: The HumanEVA dataset is used to evaluate the performance of our algorithm in terms of 3D motion annotation [1]. This dataset contains five types of motions. Based on the 16 joint coordinates in 3D space, 1590 geometric pose descriptors are extracted using the method proposed in (Chen et al. 2011) to represent 3D motion data.

– Coil20 Object dataset: We use the Coil20 dataset (Nene, Nayar, and Murase 1996) for object recognition. This dataset includes 1440 gray-scale images with 20 different objects. In our experiment, we resize each image to 32×32 and use pixel values as the features.

[1] http://vision.cs.brown.edu/humaneva/

– CMU-PIE dataset: The CMU-PIE face dataset consists of 41,368 images of 68 people. Each person was imaged under 13 different poses, 43 different illumination conditions, and with 4 different expressions. We also use the pixel values as the feature representations.

– UMIST face dataset: The UMIST face dataset consists of 564 images of 20 individuals with mixed race, gender and appearance. Each individual is shown in a range of poses from profile to frontal views. The pixel value is used as the feature representation.

Following previous works, we use the pixel value as the feature representations.

Experimental Setup

There are three parameters in our algorithms. The first one is the number of nearest neighbors and the second one is the parameter δ in Eq. (19). Following , we set the number of nearest neighbors to 5 in the experiments. The self-tune clustering method is utilized to determine the parameter δ. For the regularization parameter γ in Eq. (13) and Eq. (24), we tune them by a "grid-search" strategy from $\{10^{-6}, 10^{-4}, 10^{-2}, 10^{0}, 10^{2}, 10^{4}, 10^{6}\}$. We similarly tune the regularization parameters of all the comparison algorithms from the aforementioned range. The best results of all the comparison algorithms are reported.

Following related work, we adopt clustering accuracy (ACC) and normalized mutual information (NMI) as our evaluation metrics in our experiments.

Comparison among k-means based methods

In this section, we report the performance comparison using k-means, DisCluster, DisKmeans, AKM, HKM and Balanced k-means in terms of clustering accuracy (ACC) and NMI in Table 1.

From the experimental results, we have the following observations:

- When compared to the classical k-means clustering, DisCluster and DisKmeans algorithms, DisCluster and DisKmeans generally have better performance. This may be because discriminative dimension reduction is integrated into a single framework. Thus, each cluster is more identifiable, which helps enhance the clustering performance. We can therefore conclude that discriminative information is beneficial for clustering.
- HKM achieves the second best performance among the comparison algorithms, which indicates that most active points changing their cluster assignments at each iteration are located on or near the cluster boundaries.
- The proposed balanced k-means always gets the best performance on all the datasets. This experimental result demonstrates that the exclusive lasso is able to pose balance constraint to k-means clustering. By minimizing the exclusive lasso, the most balanced clustering result is obtained.

Table 1: Performance comparison using k-means, DisCluster, DisKmeans, AKM, HKM and Balanced k-means on nine benchmark datasets. From the experimental result, we can observe that the proposed algorithm consistently outperforms the other comparison algorithms.

	Clustering Accuracy						NMI					
	k-means	DisCluster	DisKmeans	AKM	HKM	Ours	k-means	DisCluster	DisKmeans	AKM	HKM	Ours
MNIST	52.6±3.3	53.7±2.4	54.2±3.4	52.2±3.3	55.4±3.1	**57.3±2.4**	61.7±2.5	62.5±2.8	63.1±2.6	61.9±2.1	64.3±3.1	**66.1±2.9**
USPS	65.8±2.5	67.4±2.8	70.4±2.6	66.3±2.9	71.5±2.3	**73.4±2.8**	60.8±2.3	61.4±2.5	61.9±2.1	61.0±2.6	62.5±2.2	**63.7±2.5**
YaleB	16.3±1.1	35.2±2.3	39.7±2.5	16.8±0.8	41.3±3.2	**43.5±1.8**	19.5±1.8	30.1±2.1	31.3±2.5	19.8±2.2	43.8±2.2	**46.5±2.3**
ORL	37.2±1.6	41.2±2.1	43.9±1.8	37.4±1.5	44.4±2.7	**47.2±2.2**	68.7±1.8	69.2±2.5	69.9±1.8	68.9±1.7	71.1±2.3	**73.2±2.4**
JAFFE	58.8±2.2	59.4±2.7	59.9±2.5	59.0±2.8	60.5±1.9	**61.2±1.8**	63.2±2.5	64.1±2.2	64.8±2.8	62.8±2.5	66.2±1.9	**68.4±2.2**
HuEVA	43.2±3.2	44.2±3.1	45.1±2.3	43.8±3.4	46.3±2.6	**47.7±2.5**	75.3±2.5	76.1±2.1	77.3±2.4	75.1±2.8	78.2±2.4	**79.5±2.1**
Coil20	68.4±2.8	65.3±2.6	61.3±2.3	67.9±2.7	70.3±2.4	**73.1±2.3**	59.3±2.3	60.5±2.3	61.2±2.8	59.8±2.7	63.2±2.9	**65.1±2.7**
PIE	19.5±0.8	49.8±2.7	55.5±2.9	21.2±1.1	56.1±2.2	**57.8±2.4**	24.2±2.3	25.2±2.8	25.8±2.5	24.7±1.6	57.8±2.4	**59.3±2.6**
UMIST	39.5±2.1	41.3±2.6	43.2±2.4	39.1±1.8	44.1±2.6	**46.4±2.5**	63.7±2.4	64.4±2.8	65.3±2.5	64.1±2.1	66.8±2.4	**68.1±2.3**

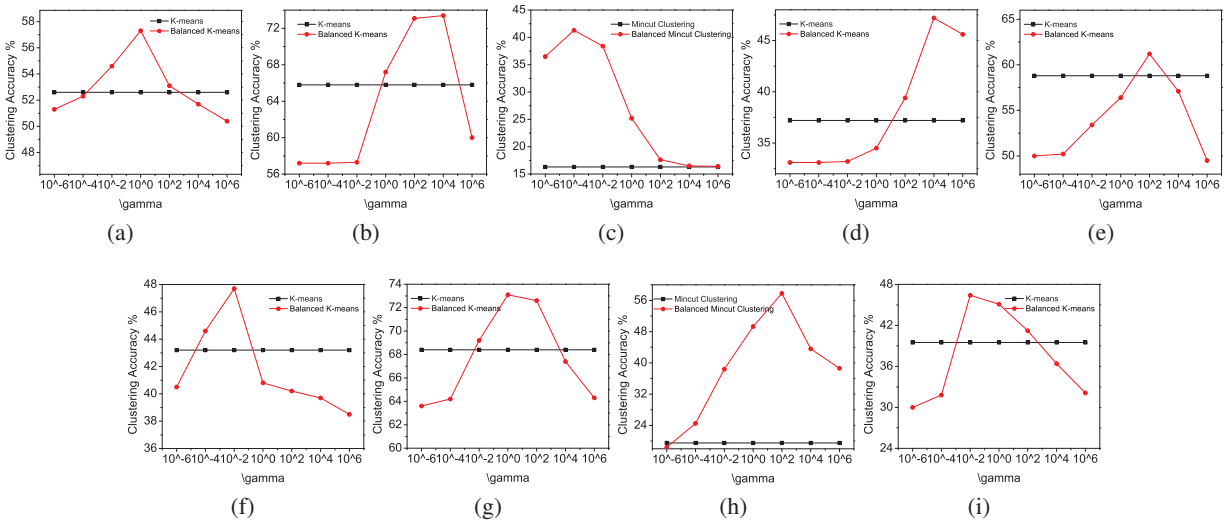

Figure 1: Parameter sensitivity of Balanced k-means. (a) MNIST (b) USPS (c) YaleB (d) ORL (e) JAFFE (f) HumanEVA (g) Coil20 (h) CMU-PIE (i) UMIST. From the results, we can observe that the parameter has a significant impact on the performance.

Comparison among graph clustering algorithms

To evaluate performance of the proposed balanced min-cut clustering algorithm, we compare it to the classical Min-Cut clustering, MinMax Cut clustering (Ding et al. 2001), Ratio Cut clustering (Hagen, , and Kahng 1992), Normalized Cut Clustering (Shi and Malik 2000) and Balanced Min-Cut clustering on the nine benchmark datasets.

We have the following observations from the experimental results reported in Table 2 and Table 3:

- Compared with the k-means based clustering, the graph clustering algorithms generally achieve better performance. This observation indicates that it is beneficial to utilize the pairwise similarities between all data points from a weighted graph adjacency matrix that contains much helpful information for clustering.
- MinMax Cut Clustering always gets the second best performance, which demonstrates that min-max clustering principle can result in more balanced partitions than the other comparison graph clustering methods.
- The proposed balanced min-cut clustering algorithm consistently outperforms the other graph clustering algorithms. From this result, we can conclude that the exclusive lasso is able to exert balance constraint on min-cut clustering and thus achieves the most balanced clustering result.

Parameter Sensitivity of the Proposed Algorithm

In this section, we study the parameter sensitivity of balanced k-means and balanced min-cut. For space limitation, we only use two datasets for example, and report the results in Figure 1 and Figure 2. From the experimental result, we can observe that γ has a significant impact on the performance of balanced K-means. We additionally show the parameter sensitivity of balanced min-cut. Similarly to

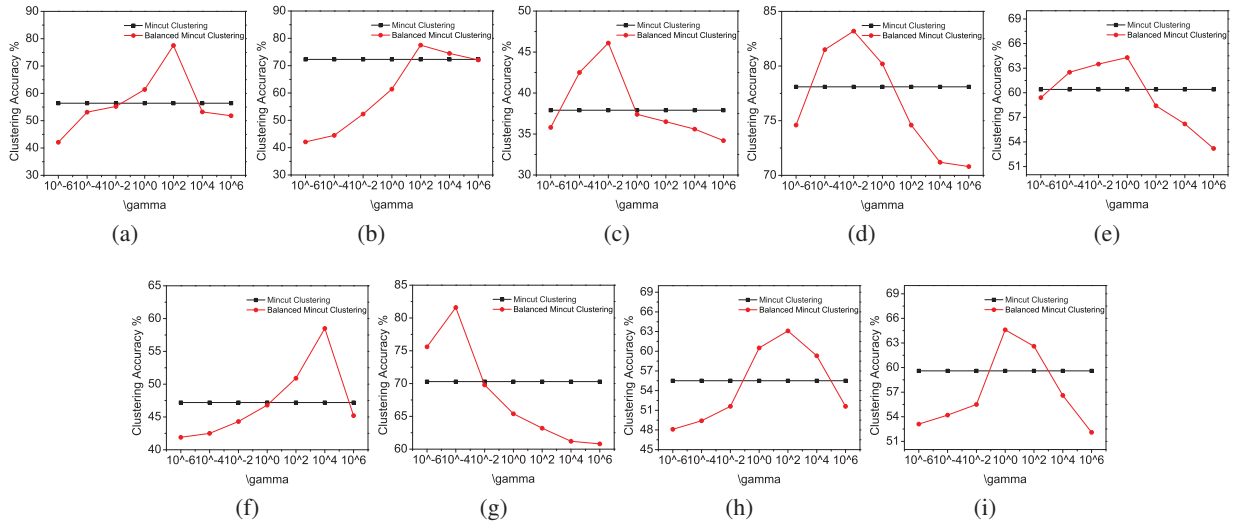

Figure 2: Parameter sensitivity of Balanced Min-Cut w.r.t γ. (a) MNIST (b) USPS (c) YaleB (d) ORL (e) JAFFE (f) HumanEVA (g) Coil20 (h) CMU-PIE (i) UMIST. From the results, we can observe that the parameter, γ has a significant impact on the performance. To be more specific, better performance is achieved when γ is in the range of $\{10^{-2}, 10^{2}\}$.

Table 2: Performance comparison in terms of clustering accuracy using the classical Min-Cut clustering, MinMax Cut clustering, Ratio Cut clustering, Normalized Cut clustering and Balanced Min-Cut clustering on nine benchmark datasets. From the experimental result, we can observe that the proposed algorithm consistently outperforms the other comparison algorithms.

	Clustering Accuracy				
	Min-Cut	Ratio Cut	Normalized Cut	MinMax Cut	Ours
MNIST	56.4±2.8	57.8±2.2	58.4±3.2	59.2±2.4	**61.4±2.0**
USPS	72.3±2.3	73.6±2.5	73.9±2.2	75.8±2.1	**77.5±2.7**
YaleB	37.9±2.8	38.2±2.4	38.6±2.1	42.2±2.6	**46.1±2.3**
ORL	45.8±1.9	46.9±2.4	47.6±2.3	48.8±1.7	**50.1±2.7**
JAFFE	60.4±2.6	61.1±2.3	62.5±2.8	62.8±2.1	**64.3±2.4**
HuEVA	47.2±2.9	48.3±2.5	48.8±2.7	49.5±3.3	**50.9±2.8**
Coil20	70.3±2.4	71.8±2.2	77.6±2.8	78.3±2.3	**81.6±1.9**
PIE	56.2±1.3	57.4±2.9	58.3±2.6	59.1±1.8	**61.3±2.8**
UMIST	59.6±2.5	60.1±2.2	60.8±2.1	62.9±1.4	**64.6±2.3**

Table 3: Performance comparison in terms of NMI using the classical Min-Cut clustering, MinMax Cut clustering, Ratio Cut clustering, Normalized Cut clustering and Balanced Min-Cut clustering on nine benchmark datasets. From the experimental result, we can observe that the proposed algorithm consistently outperforms the other comparison algorithms.

	NMI				
	Min-Cut	Ratio Cut	Normalized Cut	MinMax Cut	Ours
MNIST	65.3±2.9	66.8±2.6	67.4±3.2	68.1±2.5	**69.4±2.3**
USPS	66.5±2.3	67.9±2.5	68.4±2.9	69.8±2.7	**71.2±2.2**
YaleB	43.6±1.8	45.2±2.6	46.4±2.1	47.2±1.9	**49.1±2.4**
ORL	78.1±1.9	79.5±2.6	80.3±2.2	80.9±1.8	**83.2±2.6**
JAFFE	67.8±2.5	69.1±2.3	69.9±2.8	70.3±2.4	**73.5±1.7**
HuEVA	77.4±3.5	78.6±2.8	79.2±2.4	80.4±3.1	**82.5±2.1**
Coil20	59.8±2.9	61.4±2.3	62.7±2.5	63.6±2.8	**66.2±2.6**
PIE	55.5±2.1	61.4±2.6	62.3±2.7	62.8±2.3	**63.1±2.8**
UMIST	82.7±2.8	90.1±2.1	91.2±2.7	92.5±2.3	**94.8±2.9**

the proposed balanced k-means, the performance is heavily influenced by the parameter γ. To be more specific, better performance is usually attained when γ is in the range of $\{10^{-2}, 10^{2}\}$.

The experiments on both algorithms suggest the importance of designing an auto-tuning method for parameter selection. However, how to decide the optimal parameter is currently out of the scope in this work. We shall focus on this problem in the future.

Conclusion

In this paper, we have addressed the issue of balanced clustering which has not been studied in data mining. The exclusive lasso has been exploited to exert the balance constraint for introduce its ability to induce competition among different categories for the same data point. Particularly, we incorporated the exclusive lasso into k-means and min-cut clustering algorithms, which shall facilitate these two mainstream clustering algorithms to better cope with balanced data points. On the other hand, our objective functions are non-smooth and difficult to optimize. A new iterative approach is then designed to solve the problems. We have performed extensive experiments on a copious of datasets to evaluate performance of the proposed balanced k-means and balanced min-cut in terms of clustering accuracy and NMI. The experimental results show that our proposed algorithms

always outperform the other comparison state-of-art clustering algorithms, which validates that utilizing the exclusive lasso indeed helps achieve the most balanced clustering.

Acknowledgements

This work was particially supported by the National Natural Science Foundation of China under grant No. 61702415.

References

Chang, X.; Nie, F.; Ma, Z.; Yang, Y.; and Zhou, X. 2015. A convex formulation for shrunk spectral clustering. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, January 2530, 2015, Austin Texas, USA*.

Chen, C.; Zhuang, Y.; Nie, F.; Yang, Y.; Wu, F.; and Xiao, J. 2011. Learning a 3d human pose distance metric from geometric pose descriptor. *IEEE Trans. Visualization and Computer Graphics* 17(11).

Ding, C., and Li, T. 2007. Adaptive dimension reduction using discriminant analysis and k-means clustering. In *Proc. ICML*, 521–528.

Ding, C. H.; He, X.; Zha, H.; Gu, M.; and Simon, H. D. 2001. A min-max cut algorithm for graph partitioning and data clustering. In *Proc. ICDM*.

Filippone, M.; Camastra, F.; Masulli, F.; and Rovetta, S. 2008. A survey of kernel and spectral methods for clustering. *Pattern recognition* 41(1):176–190.

Georghiades, A. S.; Belhumeur, P. N.; and Kriegman, D. 2001. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. PAMI* 23(6):643–660.

Girolami, M. 2002. Mercer kernel-based clustering in feature space. *IEEE Trans. Neural Networks* 13(3):780–784.

Hagen, L.; ; and Kahng, A. B. 1992. New spectral methods for ratio cut partitioning and clustering. *Trans. Computer-aided design of integrated circuits and systems*.

Jain, A. K., and Dubes, R. C. 1988a. *Algorithms for clustering data*. Prentice-Hall, Inc.

Jain, A. K., and Dubes, R. C. 1988b. Algorithms for clustering data. *Englewood Cliffs,NJ: Prentice-Hall*.

Jain, A. K.; Murty, M. N.; and Flynn, P. J. 1999. Data clustering: a review. *ACM computing surveys* 31(3):264–323.

la Torre, D.; Fernando; and Kanade, T. 2006. Discriminative cluster analysis. In *Proc. ICML*, 241–248.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 2011. Gradient-based learning applied to document recognition. *Proc. IEEE*.

Li, Z.; Nie, F.; Chang, X.; and Yang, Y. 2017. Beyond trace ratio: Weighted harmonic mean of trace ratios for multiclass discriminant analysis. *IEEE Trans. Knowl. Data Eng.* 29(10):2100–2110.

Lyons, M. J.; Budynek, J.; and Akamatsu, S. 1999. Automatic classification of single facial images. *IEEE Trans. PAMI* 21(12):1357–1362.

Nene, S. A.; Nayar, S. K.; and Murase, H. 1996. Columbia object image library (coil-20). Technical report, CUCS-005-96, Columbia University.

Nie, F.; Cai, G.; and Li, X. 2017. Multi-view clustering and semi-supervised classification with adaptive neighbours. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, 2408–2414.

Nie, F.; Xu, D.; and Li, X. 2012. Initialization independent clustering with actively self-training method. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 42(1):17–27.

Raj, A., and Wiggins, C. H. 2010. An information-theoretic derivation of min-cut-based clustering. *IEEE PAMI* 32(6).

Saha, B., and Mitra, P. 2006. Fast incremental minimum-cut based algorithm for graph clustering. *Proc. ICDM* 207–211.

Samaria, F. S., and Harter, A. C. 1994. Parameterisation of a stochastic model for human face identification. In *Proc. Applications of Computer Vision*, 138–142.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *Trans. PAMI* 888–905.

Turaga, D. S.; Vlachos, M.; and Verscheure, O. 2009. On k-means cluster preservation using quantization schemes. *Proc. ICDM*.

Wang, X., and Davidson, I. 2010. Flexible constrained spectral clustering. In *Proc. KDD*, 563–572.

Wang, J.; Wang, J.; Ke, Q.; Zeng, G.; and Li, S. 2012. Fast approximate k-means via cluster closures. In *Proc. CVPR*.

Wang, C.; She, Z.; and Cao, L. 2013. Coupled clustering ensemble: Incorporating coupling relationships both between base clusterings and objects. In *Proc. ICDE*.

Wu, L.; Hoi, S. C.; Jin, R.; Zhu, J.; and Yu, N. 2012a. Learning bregman distance functions for semi-supervised clustering. *IEEE Trans. Knowledge and Data Eng.* 24(3):478–491.

Wu, O.; Hu, W.; Maybank, S. J.; Zhu, M.; and Li, B. 2012b. Efficient clustering aggregation based on data fragments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 42(3):913–926.

Ye, J.; Zhao, Z.; and Liu, H. 2007a. Adaptive distance metric learning for clustering. In *Proc. CVPR*, 1–7.

Ye, J.; Zhao, Z.; and Liu, H. 2007b. Adaptive distance metric learning for clustering. In *Proc. CVPR*, 1–7.

Ye, J.; Zhao, Z.; and Wu, M. 2007. Discriminative k-means for clustering. In *Proc. NIPS*, 1649–1656.

Yuan, J.; Bae, E.; Tai, X.-C.; and Boykov, Y. 2014. A spatially continuous max-flow and min-cut framework for binary labeling problems. *Numerische Mathematik*.

Zhang, C., and Xia, S. 2009. K-means clustering algorithm with improved initial center. In *Proc. Knowledge Discovery and Data Mining*, 790–792.

Zhou, Y.; Jin, R.; and Hoi, S. 2010. Exclusive lasso for multi-task feature selection. *Proc. ICAIS* 988–995.