Balanced Clustering with Least Square Regression

Hanyang Liu,[1] **Junwei Han,**[1*] **Feiping Nie,**[2*] **Xuelong Li**[3]
[1]School of Automation, Northwestern Polytechnical University, Xi'an, 710072, P. R. China
[2]School of Computer Science and Center for OPTIMAL, Northwestern Polytechnical University, Xi'an, 710072, P. R. China
[3]Center for OPTIMAL, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics
and Precision Mechanics, Chinese Academy of Sciences, Xi'an, 710119, Shaanxi, P. R. China
{ericstarkhan, junweihan2010, feipingnie}@gmail.com, xuelong_li@ieee.org

Abstract

Clustering is a fundamental research topic in data mining. A balanced clustering result is often required in a variety of applications. Many existing clustering algorithms have good clustering performances, yet fail in producing balanced clusters. In this paper, we propose a novel and simple method for clustering, referred to as the Balanced Clustering with Least Square regression (BCLS), to minimize the least square linear regression, with a balance constraint to regularize the clustering model. In BCLS, the linear regression is applied to estimate the class-specific hyperplanes that partition each class of data from others, thus guiding the clustering of the data points into different clusters. A balance constraint is utilized to regularize the clustering, by minimizing which can help produce balanced clusters. In addition, we apply the method of augmented Lagrange multipliers (ALM) to help optimize the objective model. The experiments on seven real-world benchmarks demonstrate that our approach not only produces good clustering performance but also guarantees a balanced clustering result.

Introduction

Clustering has been widely studied for decades and plays an essential role in many fields, such as statistics and artificial intelligence. The objective of clustering is to group the data points that have similar patterns into the same cluster and discover the data structure. Over the past decades, many clustering algorithms have been proposed and extended, such as K-means, fuzzy C-means (Bezdek 2013), spectral clustering methods (Ng et al. 2002; Nie, Wang, and Huang 2016), and projected clustering (Nie, Wang, and Huang 2014).

Given data points with balanced distribution (each class has approximately the same number of samples), usually we would expect the clustering result to reflect such balance. In other words, a good clustering algorithm is supposed to prevent a too small or too great number of data points from being partitioned into a cluster. Nevertheless, prevalent clustering algorithms like K-means, spectral clustering, and etc., do not produce a balanced clustering result, especially when the data points need to be grouped into a large number of clusters.

In many data mining applications, it is often required to have balanced clusters, and ordinary clustering algorithms are unable to meet the requirement. Examples can be found in photo query systems (Althoff, Ulges, and Dengel 2011) and retail chain problems. Another promising application is in the energy load balance of wireless sensor networks (Du, Liu, and Qian 2009), where unbalanced cluster structure may cause unbalanced energy consumption and shorten the network lifetime. Moreover, balanced clustering tends to avoid forming outlier clusters, and thus has beneficial regularizing effect (Zhong and Ghosh 2003). Despite the wide and essential application of balanced clustering, it seems to have drawn no continuous attention from the community.

A few early proposed clustering algorithms are able to produce balanced clusters. These balanced algorithms can be categorized into two types: a) hard-balanced clustering, in which cluster size is strictly required by setting fixed number of samples in clusters; b) soft-balanced clustering, in which balance is an aim but not a mandatory requirement. Constraint K-means (Bradley, Bennett, and Demiriz 2000) and a lately proposed method, balanced K-means (Malinen and Fränti 2014) are based on K-means and the number of data points in clusters is set as a parameter. They are both hard-balanced clustering. Actually in many situations, absolute balance is not required. The method in (Banerjee and Ghosh 2002) is based on a three step sampling procedure, and their subsequent work (Banerjee and Ghosh 2004) utilizes the penalty strategy to increase the distance from the data points to the centroid that has already won data points. In the works (Zhong and Ghosh 2003; Chang et al. 2014) the balance degree can be adjusted. These algorithms are all soft-balanced clustering methods.

Balance is a global property, therefore it is very difficult to guarantee both a balanced result and high cluster quality. In our work, we aim to integrate the two goals, rather than compromise either one. We consider a balance constraint to regularize the clustering model, which belongs to the soft-balanced algorithms, in order to have a balanced result and maintain good clustering performance simultaneously. This is the first motivation of our proposed method.

Linear regression plays an essential role in supervised learning tasks, such as classification, and demonstrates tremendously excellent performance as a learning model in processing high dimensional data, such as in face recogni-

[*] Corresponding authors.

tion (Chen 2014; Tahir et al. 2011). Previous works (Nie et al. 2009; 2011) and (Ye 2007) reveal the secret of this property: the true data assignment matrix can be always embedded into a low dimensional linear mapping of the data, generally when the data are high-dimensional and small-sample-sized. Also, we noted that the linear regression can provide the model of dissimilarity for clustering to guide the partitioning. From this perspective, we propose a clustering model with minimized least square error of linear regression. The third part of this paper shows that the linear regression can estimate the class-specific hyperplanes dividing each class of data from others. This the second motivation of our work.

Contributions. Driven by the aforementioned two motivations, we propose a novel clustering algorithm, based on the least square linear regression with a balance constraint. The proposed clustering method is called balanced clustering with least square regression (BCLS). To efficiently minimize the objective function and obtain a better solution, we apply the augmented Lagrange multipliers (ALM) for the optimization problem. We evaluate our proposed method on seven real-world datasets and it turns out that this clustering strategy works well in clustering different types of data, and demonstrates excellent balancing performance.

Background

In this section, we briefly introduce two key background methods we use in our proposed model. We build our clustering model based on the least square linear regression and then apply the method of ALM to tackle the optimization problem.

We first introduce some notations that are used throughout the paper. For matrix $M = (m_{ij}) \in \mathbb{R}^{p\times q}$, m_{ij} denotes the (i, j)-th entry of M, M^T denotes the transpose of M, and $\mathrm{tr}(M)$ denotes the trace of M. The F-norm of M is denoted by $\|M\|_F$, and the l_2-norm is denoted by $\|M\|_2$. The inner product of matrices A and B is denoted by $\langle A{\cdot}B\rangle$. $\mathbf{1}$ denotes the vector with all elements as 1, and $\mathbf{0}$ denotes the vector with all elements as 0.

Least Square Linear Regression

Linear regression is a traditional approach for regression problems. This antique, yet efficient method has been widely used in real-world applications. Given a dataset of two classes, $\{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$, and $y_i \in \{-1, 1\}$ is the label of the i-th data point, the linear regression model has the following form:

$$f(x) = x^T w + b \tag{1}$$

where $w \in \mathbb{R}^d$ is the projection vector, and $b \in \mathbb{R}$ is the bias of the linear model. In regression, the estimation error is minimized as follows:

$$\min \sum_{i=1}^{n} e(f(x_i), y_i) \tag{2}$$

In the multivariate linear regression (Trevor, Robert, and Jerome 2001), we are given a dataset of c classes, $X = [x_1, x_2, \cdots, x_n] \in \mathbb{R}^{d\times n}$, where n is the number of samples and d is the data dimensionality, and $Y = [y_1, y_2, \cdots, y_n]^T \in \mathbb{R}^{n\times c}$ is the corresponding label matrix. The least square error is a popular approach for the linear regression model, aiming to obtain the optimal projection matrix $W = (w_{ik}) \in \mathbb{R}^{d\times c}$, and the bias vector $b \in \mathbb{R}^c$ by the following optimization model

$$\min_{W,b} \sum_{i=1}^{n} \|W^T x_i + b - y_i\|_2^2 + \gamma R(W) \tag{3}$$

A regularization term $R(W)$ with a coefficient γ is introduced as a penalty term for the size of W.

The Method of Augmented Lagrange Multipliers

Augmented Lagrangian methods are a series of algorithms for solving constrained optimization problems. The later work (Bertsekas 1982) introduced the general method of augmented Lagrange multipliers (ALM) for solving constraint optimization problem of the following kind:

$$\min f(X), \;\; s.t.\; H(X) = \mathbf{0} \tag{4}$$

where $f : \mathbb{R}^{m\times n} \to \mathbb{R}$ and $H : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$. The augmented Lagrangian function is defined as

$$L(X, \Lambda, \mu) = f(X) + \langle \Lambda, H(X)\rangle + \frac{\mu}{2}\|H(X)\|_F^2 \tag{5}$$

where μ is a positive scalar that gets updated after each iteration, and Λ is an estimate of the Lagrange multi-plier, with the estimation accuracy improved at every step.

Compared to common used penalty methods, the major advantage of ALM method lies on solving the original constraint problem without taking $\mu \to \infty$ and that μ can stay much smaller when the objective function converges.

Hence, the constraint optimization problem becomes an unconstraint problem by minimizing the Lagrangian function $L(X, \Lambda, \mu)$ with updating parameters μ and Λ. The indication of convergence is $H(X) \to \mathbf{0}$ or Λ remains unchanged. The detailed algorithm is shown in Algorithm 1.

Proposed Model

Balance Constraint

In our work, we consider a common used class indicator matrix $Y = (y_{ik}) \in \mathbb{R}^{n\times c}$, following the setting in (Trevor, Robert, and Jerome 2001), as the clustering assignment matrix. The assignment matrix is defined as follows:

$$y_{ik} = \begin{cases} 1, & \text{if } x_i \in \text{class } k \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

For convenience, we simply express the assignment matrix as $Y \in Ind$.

Our goal is to partition the data samples into balanced clusters among different categories, preventing that any cluster has too big or small number of samples. Given Y as the assignment matrix, and $s = [s_1, s_2, \cdots, s_c] \in \mathbb{R}^{1\times c}$, where s_i denotes the number of samples in the i-th cluster. Apparently we have $s = \mathbf{1}^T Y$, and the average number of the samples in each cluster is n/c.

Algorithm 1 Algorithm of General Method of ALM

```
Input: $\rho > 1$
Output: Solution $X$
Initialize $\mu$, and set $\Lambda = \mathbf{0}$
while not converge do
    1. Solve $X^{(t+1)} = \arg\min_X L(X^{(t)}, \Lambda^{(t)}, \mu^{(t)})$ in Eq.(5);
    2. Update $\Lambda$: $\Lambda^{(t+1)} = \Lambda^{(t)} + \mu^{(t)} H(X^{(t+1)})$;
    3. Update $\mu$: $\mu^{(t+1)} = \rho\mu^{(t)}$.
end while
Return $X$.
```

To partition all the samples into balanced clusters means to make the cluster size as close to n/c as possible. That is to say, our purpose is to minimize σ^2, the variance of $\{s_k\}$:

$$\begin{aligned}\min_s \sigma^2 \quad &\Leftrightarrow \quad \min_s \frac{1}{c}\sum_{k=1}^{c}\left(s_k - \frac{n}{c}\right)^2 \\ &\Leftrightarrow \quad \min_s \sum_{k=1}^{c}\left(s_k^2 - 2s_k\frac{n}{c} + \frac{n^2}{c^2}\right) \\ &\Leftrightarrow \quad \min_s \left(\sum_{k=1}^{c} s_k^2 - \frac{n^2}{c}\right) \\ &\Leftrightarrow \quad \min_s \sum_{k=1}^{c} s_k^2 \end{aligned} \tag{7}$$

With simple mathematical deduction, we can get

$$\sum_{k=1}^{c} s_k^2 = \|s\|_2^2 = \left\|\mathbf{1}^T Y\right\|_2^2 = \operatorname{tr}\left(Y^T \mathbf{1}\mathbf{1}^T Y\right) \tag{8}$$

From Eq. (7) and Eq. (8), we can observe that we can achieve the goal of balanced clustering by minimizing the square-sum of the number of samples in each cluster. Following this idea, we use $\operatorname{tr}(Y^T\mathbf{1}\mathbf{1}^T Y)$ as a balance constraint, and it is obvious that the its value indicates the balance degree of our clustering algorithm.

Theorem 1. *Given $s_1 + s_2 + \cdots + s_c = n$ and $s_k|_{k=1}^c \geq 0$,$\sum_{k=1}^c s_k^2$ reaches its minimal value n^2/c when $s_k = n/c$.*

***Proof*.** According to the Cauchy-Schwarz Inequality,

$$\left(\sum_{k=1}^{c} s_k t_k\right)^2 \leq \left(\sum_{k=1}^{c} s_k^2\right)\left(\sum_{k=1}^{c} t_k^2\right) \tag{9}$$

Let $t_k|_{k=1}^c = 1$, the equality holds when $s_1 = s_2 = \cdots = s_c$. So we can readily conclude that when $s_k|_{k=1}^c = n/c$, $\sum_{k=1}^c s_k^2$ arrives at its minimal value n^2/c. □

According to the theorem above, the balance constraint is capable of introducing competition among different classes. By minimizing the balance constraint $\operatorname{tr}(Y^T\mathbf{1}\mathbf{1}^T Y)$, the data samples tend to be clustered into c balanced classes with n/c samples in each class.

Algorithm 2 Algorithm of the BCLS method

```
Input: Centered dataset $X = [x_1, x_2, \cdots, x_n] \in \mathbb{R}^{d\times n}$,
       the number of clusters $c$, and parameters $\gamma$, $\lambda$, and
       $\rho(\rho > 0)$.
Output: Assignment matrix $Y \in \mathbb{R}^{n\times c}$.
Initialize $Y$ randomly, initialize $\mu$, and set $\Lambda = \mathbf{0} \in \mathbb{R}^{n\times c}$.
Repeat
    1. Obtain $W$ and $b$ by solving the problem in Eq. (14) with
       the solution in Eq. (15);
    2. Obtain $Z$ by solving Eq. (16) with the solution in Eq.
       (17);
    3. Obtain $Y$ by Eq. (19) and Eq. (21);
    4. Update $\Lambda$: $\Lambda^{(t+1)} = \Lambda^{(t)} + \mu^{(t)}(Y - Z)$;
    5. Update $\mu$: $\mu^{(t+1)} = \rho\mu^{(t)}$.
Until $Y - Z \to \mathbf{0}$ or $\Lambda$ remains unchanged
Return $Y$.
```

Objective Function

Linear regression models have been widely used and have excellent performance in supervised learning, such as classification. Few previous research has considered the application of regression model for unsupervised learning. Unlike the previous works, we proposed a novel clustering model based on the least square linear regression.

In our clustering setting, given n data samples of c classes, each column of X stands for a sample that consists of d features. For simplicity, we assume the data are centered, i.e., $X\mathbf{1} = \mathbf{0}$. We combine the clustering goal with the regression model in Eq. (3), and adopt the l_2-regularization on the projection matrix W as the penalty for the size of W, as in (Zou, Hastie, and Tibshirani 2006). Besides, based on the observation in the previous subsection, we introduce the balance constraints $\operatorname{tr}(Y^T\mathbf{1}\mathbf{1}^T Y)$ as another regularization term in our clustering model. With the setting above, we proposed the BCLS with the following objective function:

$$\begin{aligned}\min_{W,b,Y\in Ind} &\left\|X^T W + \mathbf{1}b^T - Y\right\|_F^2 + \gamma\|W\|_F^2 \\ &+ \lambda\operatorname{tr}\left(Y^T\mathbf{1}\mathbf{1}^T Y\right)\end{aligned} \tag{10}$$

where γ is the regularization parameter and λ is the balance parameter.

Our BCLS method is based on the linear regression model, aiming to estimate in each iteration, the class-specific regression hyperplanes that partition the data of distinct classes. From this perspective, we can view the projection matrix $W = [w_1, w_2, \cdots, w_c] \in \mathbb{R}^{d\times c}$ as the catenation of $\{w_k\}$, where w_k denotes the normal vector to the hyperplane that partitions the k-th class from the others. The accuracy of the estimation of these hyperplanes gets improved step by step, in the process of minimizing the least square regression error.

In the objective function in Eq. (10), the minimization of the least square regression term guides the clustering process to partition data points into c clusters. Meanwhile, minimizing the balance regularization term guarantees the balanced partitioning among different categories.

Optimization Algorithm

In this section, we give an algorithm to optimize the objective function of BCLS in Eq. (10). Since the problem is NP-hard in general, it is very hard to solve it in polynomial time. In our work, we apply ALM to help obtain good solutions for the optimization. We replace the assignment matrix Y in the balance term with matrix Z that has entries with continuous values, following the alternating direction method of multipliers (Eckstein and Bertsekas 1992), so as to transfer Eq. (10) into an equality constraint optimization problem and approximately obtain the optimal solution by alternatively solving Y with Z fixed and solving Z with Y fixed. Then Eq. (10) becomes

$$\min_{\substack{W,b,Y\in Ind\\ Y=Z}} \left\|X^TW+\mathbf{1}b^T-Y\right\|_F^2+\gamma\|W\|_F^2 \\ +\lambda\mathrm{tr}\left(Z^T\mathbf{1}\mathbf{1}^TZ\right) \tag{11}$$

Now we can adopt ALM to solve the above optimization problem with equality constraint $H(X)=Y-Z=0$. In the method of ALM, the goal is to minimize the Lagrangian function shown in Eq. (5). With simple mathematical deduction, the optimization problem of the general method of ALM can be converted to:

$$\min_X f(X)+\frac{\mu}{2}\left\|H(X)+\frac{1}{\mu}\Lambda\right\|_F^2 \tag{12}$$

We apply the method of ALM above, to the optimization problem in Eq. (11), and we get the final optimization problem for our BCLS method:

$$\min_{W,b,Y\in Ind,Z} \left\|X^TW+\mathbf{1}b^T-Y\right\|_F^2+\gamma\|W\|_F^2 \\ +\lambda\mathrm{tr}\left(Z^T\mathbf{1}\mathbf{1}^TZ\right)+\frac{\mu}{2}\left\|Y-Z+\frac{1}{\mu}\Lambda\right\|_F^2 \tag{13}$$

where μ is a positive scalar and its value increases slightly after each iteration, and Λ is an estimate of the Lagrange multiplier. The estimation accuracy improves at every step of iteration.

The problem in Eq. (13) is non-convex, and the objective function has multiple unknown variables. It can be solved by alternatively updating the four variables W, b, Y and Z.

i) With Y and Z fixed, Eq. (13) becomes

$$\min_{W,b} \left\|X^TW+\mathbf{1}b^T-Y\right\|_F^2+\gamma\left\|W\right\|_F^2 \tag{14}$$

Noting that the data are centered, simply by setting the derivatives of the objective function in Eq. (14) with respect to W and b to zeros, we have

$$\begin{cases} W=\left(XX^T+\gamma I_d\right)^{-1}XY \\ \;\;b=\dfrac{1}{n}Y^T\mathbf{1} \end{cases} \tag{15}$$

ii) With W, b and Y fixed, Eq. (13) becomes

$$\min_Z \lambda\mathrm{tr}\left(Z^T\mathbf{1}\mathbf{1}^TZ\right)+\frac{\mu}{2}\left\|Y-Z+\frac{1}{\mu}\Lambda\right\|_F^2 \tag{16}$$

Table 1: Description of Benchmark Datasets

Dataset	# Sample	# Dimension		# Class
		Original	Processed	
Wine	144	13	13	3
Ionosphere	252	34	20	2
UMIST	380	10304	50	20
YALE-B	2242	1024	70	38
AR	1400	4800	125	100
JAFFE	200	4096	20	10
CMU-PIE	1000	4096	50	10

Likewise, we can obtain Z by setting the derivative of the objective function in Eq. (16) with respect to Z to zero:

$$Z=\left(\mu I_n+2\lambda\mathbf{1}\mathbf{1}^T\right)^{-1}\left(\mu Y+\Lambda\right) \tag{17}$$

iii) With W, b and Z fixed, we can transform Eq. (13) into the following form:

$$\min_{Y\in Ind} \|Y-V\|_F^2+const. \tag{18}$$

where $const.$ denotes a constant and $V=(v_{ik})\in\mathbb{R}^{n\times c}$, and

$$V=\frac{2}{2+\mu}\left(X^TW+\mathbf{1}b^T\right)+\frac{1}{2+\mu}(\mu Z-\Lambda) \tag{19}$$

Considering $Y\in Ind$, each element of $Y=(y_{ik})\in\mathbb{R}^{n\times c}$ is binary and the sum of each row is 1, thus Eq. (18) can be written as:

$$\min_Y \sum_{i=1}^n\sum_{k=1}^c(y_{ik}-v_{ik})^2, s.t.\; y_{ik}\in\{0,1\}, \sum_{k=1}^c y_{ik}=1 \tag{20}$$

We use a traversal strategy to solve Eq. (20), and consider the locations of 1 in Y from one row to another. We can obtain the solution as follows

$$y_{ik}=\begin{cases} 1, \text{ if } k=\arg\max\limits_k\{v_{ik}\}_{k=1}^c \\ 0, \text{ otherwise} \end{cases} \tag{21}$$

The detailed algorithm of BCLS is shown in Algorithm 2.

Algorithm Analysis. Since the problem in Eq. (13) is non-convex, given fixed Λ and μ, Algorithm 2 will find the local solution in each iteration. The convergence of ALM has been analyzed and proven in previous papers (Bertsekas 1982; Powell 1969). In each iteration, the major computation burden seems to lie on obtaining the matrix inverse in Step 1 and Step 2. Actually, the inverse in Step 1 can be computed before the iteration. Besides, the matrix $\mu I_n+2\lambda\mathbf{1}\mathbf{1}^T$ in Step 2 is very special, and its inverse can be given by $\left[(\mu+2n\lambda)I_n-2\lambda\mathbf{1}\mathbf{1}^T\right]/(\mu^2+2n\lambda\mu)$. Hence the time complexity in a single iteration is $O(n^2c+d^2c)$.

Experiment

In this section, we evaluate the clustering and balancing performance of the proposed method on benchmark datasets. We compare our method with K-means (KM), fuzzy C-means (FCM) (Bezdek 2013), Discriminative K-means (DKM) (Ye, Zhao, and Wu 2008), Spectral Clustering (SC) (Chen et al. 2011), and Balanced K-means (BKM)

Table 2: Clustering performance (evaluated by ACC and NMI) and balancing performance (evaluated by $\mathrm{N_{entro}}$) of KM, SC, FCM, DKM, BKM, and our method BCLS. Reported are the ACC, NMI and $\mathrm{N_{entro}}$ score of these methods on each dataset corresponding to the best objective function values over 20 random initializations.

Metric	Dataset	KM	SC	FCM	DKM	BKM	BCLS
ACC	Wine	96.53	97.22	96.53	**98.61**	95.83	**98.61**
	Ionosphere	76.59	75.40	76.59	76.59	74.60	**77.78**
	UMIST	58.68	**66.84**	58.68	64.21	61.05	66.58
	YALE-B	11.60	27.83	11.51	42.24	10.35	**43.44**
	AR	25.07	35.43	30.57	59.50	29.29	**65.86**
	JAFFE	89.00	97.00	96.50	90.00	96.00	**100**
	CMU-PIE	20.30	34.30	21.50	45.00	22.20	**73.80**
NMI	Wine	86.71	89.70	86.71	93.83	83.37	**93.85**
	Ionosphere	21.53	19.53	21.53	21.53	18.25	**23.58**
	UMIST	72.23	**78.37**	70.83	75.57	72.76	74.22
	YALE-B	16.27	33.36	15.61	53.99	14.83	**54.50**
	AR	60.91	63.12	61.67	80.07	61.46	**80.79**
	JAFFE	90.77	96.95	96.08	89.31	95.65	**100**
	CMU-PIE	10.88	33.15	12.88	45.74	13.93	**67.78**
$\mathrm{N_{entro}}$	Wine	0.9991	0.9983	0.9991	0.9996	**1**	**1**
	Ionosphere	0.9989	0.9993	0.9989	0.9989	**1**	**1**
	UMIST	0.9771	0.9874	0.9814	0.9743	**1**	**0.9999**
	YALE-B	0.9724	0.9394	0.9817	0.9224	**1**	**0.9997**
	AR	0.9635	0.9619	0.9835	0.9414	**1**	**0.9991**
	JAFFE	0.9672	0.9960	0.9959	0.9863	**1**	**1**
	CMU-PIE	0.9598	0.9443	0.9776	0.8195	**1**	**0.9999**

(Malinen and Fränti 2014). Specifically, KM and SC are among the most classical algorithms that have been widely used for years, FCM and DKM are two efficient K-means-like methods, and BKM is a state-of-the-art hard-balanced clustering method based on K-means. We choose several K-means-like methods for comparison due to the linear property of both K-means and our method.

Experiment Setup

Datasets and Preprocessing Seven real-world datasets are used in the experiments, including two UCI datasets, Wine and Ionosphere[*], and five face datasets, UMIST[**], YALE-B, AR, JAFFE[†], and CMU-PIE.

For the high dimensional datasets, dimension reduction is performed in the preprocessing of the datasets. We apply PCA with 80% to 95% of the information reserved on all the datasets except Wine. Moreover, to better evaluate the balancing capacity of each algorithm, we resize some of the datasets, making each dataset has equal number of samples in every classes. The detailed information of the processed datasets is shown in Table 1.

Parameter Settings There are four parameters in the BCLS. The first one is the regularization parameter γ in Eq.(13), which is essential but the its value does not sensitively influence the performance. We set γ to 10^{-5} for all the datasets. The balance parameter λ, and the coefficient of the Lagrange multipliers μ, play very important roles in the BCLS algorithm and both should be determined carefully. We tune them by grid search from $\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}\}$ and $\mu \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$. The last one is ρ, the updating rate of μ, and it should be set slightly greater than 1 (Bertsekas 1982). We set ρ to 1.002 for AR and 1.005 for the rest. For all the other algorithms, we also tune the parameters carefully and report the results under the best parameter settings.

The experiment results of all the clustering algorithms we discuss here depend on the initialization. To reduce the statistical variation, we independently repeat all the clustering algorithms for 20 times with random initialization. We report the clustering result of each algorithm corresponding to its best objective function value, respectively.

Evaluation We adopt the common used metrics, Clustering Accuracy (ACC) (Cai, He, and Han 2005) and Normalized Mutual Information (NMI) (Strehl and Ghosh 2002) to evaluate the clustering performance for all the algorithms. We also use the Normalized Entropy ($\mathrm{N_{entro}}$) (Zhong and Ghosh 2003) to evaluate their balancing performance:

$$\mathrm{N_{entro}} = -\frac{1}{\log c} \sum_{k=1}^{c} \frac{n_k}{n} \log \frac{n_k}{n} \tag{22}$$

where n_k is the number of data objects in cluster k. An $\mathrm{N_{entro}}$ of 1 means perfectly balanced clusters and 0 means extremely unbalanced clusters.

Experiment Results

The clustering performance (ACC and NMI score) and balancing performance ($\mathrm{N_{entro}}$ score) of each algorithm on the

[*] https://archive.ics.uci.edu/ml/index.html

[**] http://images.ee.umist.ac.uk/danny/database.html

[†] http://www.kasrl.org/jaffe.html

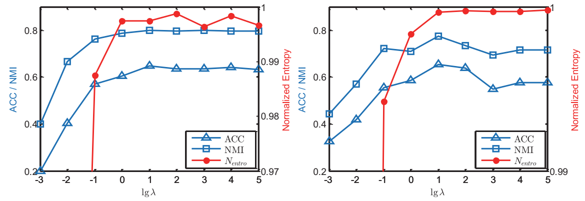


Figure 1: The effect of the balance parameter in BCLS on ACC. Left: **AR** dataset; Right: **UMIST** dataset.

seven datasets are shown in Table 2. We plot the data points distribution on the UMIST dataset, which is chosen arbitrarily due to the limit space of the paper, as shown in Figure 2. Moreover, the effect of the balance parameter λ in BCLS on the clustering and balancing performance is demonstrated in Figure 1, where we take the datasets AR and UMIST as examples. We have the observations as follows:

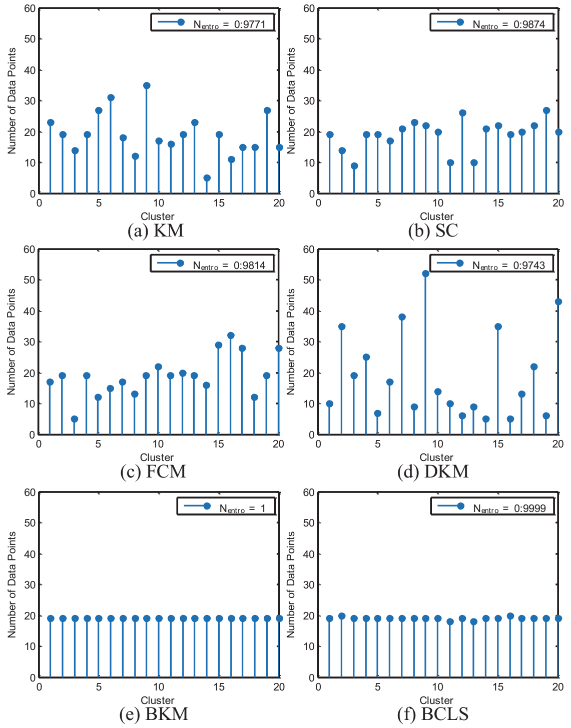
- The proposed algorithm demonstrates the best clustering performance on most of the datasets except UMIST. It even reaches the ACC and NMI score of 100% on the dataset JAFFE.

- Each algorithm shows similar clustering performance on the relatively low dimensional datasets (e.g., Wine and Ionosphere). SC considers the geometry structure of the data, hence relatively has an advantage in clustering high dimensional data, compared to KM and FCM.

- The proposed algorithm significantly has better clustering performance on the face datasets (e.g., CMU-PIE and YALE-B), due to the good capacity of linear regression in dealing with high dimensional data (Chen 2014; Tahir et al. 2011).

- Our algorithm outperforms other algorithms except BKM, in balancing performance. BKM is based on the hard-balanced strategy and produce strictly balanced clusters all the time, so its $\mathrm{N_{entro}}$ score on all the datasets consistently remains 1. Figure 2 shows that BKM and BCLS produce balance clusters, while the clusters of the other algorithms remain unbalanced (From Figure 2 we observe that, a slight numerical reduction on $\mathrm{N_{entro}}$ indicates big sample number fluctuation among clusters).

- The balance parameter λ in BCLS has evident influence on the clustering and balancing performance. From Figure 1 we can observe that, with the increasing value of λ, both the ACC and NMI scores rapidly reach their maximum values respectively and then slightly decrease. The clusters are generally more balanced with a greater value of λ. We also observe that the balance constraint helps evenly distribute the data points, and turns out increasing the clustering performance. Hence, the best clustering result with balanced clusters can always be achieved by incorporating the least square regression term and the balance term, with a proper balance parameter λ.



(a) KM

(b) SC

(c) FCM

(d) DKM

(e) BKM

(f) BCLS

Figure 2: Sample distribution of clusters corresponding to the best objective function values on UMIST: (a) **KM**; (b) **SC**; (c) **FCM**; (d) **DKM**; (e) **BKM**: BKM is a hard-balanced algorithm producing absolutely balanced clusters; and (f) **BCLS**: outperforms all other algorithms in balancing performance, except BKM with hard-balanced strategy.

Conclusion

In this paper, we proposed a conceptually simple but effective clustering algorithm that produces balanced clusters. We estimate the class-specific hyperplanes that partition the data points into different clusters by iteratively minimizing the least square error of the linear regression. In our proposed method, a balance constraint was used to regularize the clustering model, in order to achieve a balanced clustering result. Moreover, we applied ALM in the optimization to obtain good solutions for the problem. The experiments on seven real-world benchmark datasets demonstrated that the proposed algorithm BCLS produces good clustering and balancing performances simultaneously. In the future study, we may aim to apply BCLS into practical use such as saliency detection (Han et al. 2015), remote sensing (Cheng et al. 2015; Lu, Wu, and Yuan 2014), and image super-resolution (Lu, Yuan, and Yan 2013).

Acknowledgments

This work was supported in part by the National Science Foundation of China under Grant No.61473231 and Grant No.61522207.

References

Althoff, C. T.; Ulges, A.; and Dengel, A. 2011. Balanced clustering for content-based image browsing. *Series of the Gesellschaft Fur Informatik* 27–30.

Banerjee, A., and Ghosh, J. 2002. On scaling up balanced clustering algorithms. In *SDM*, volume 2, 333–349. SIAM.

Banerjee, A., and Ghosh, J. 2004. Frequency-sensitive competitive learning for scalable balanced clustering on high-dimensional hyperspheres. *Neural Networks, IEEE Transactions on* 15(3):702–719.

Bertsekas, D. P. 1982. *Constrained Optimization and Lagrange Multiplier Methods*. Academic press.

Bezdek, J. C. 2013. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Springer Science & Business Media.

Bradley, P.; Bennett, K.; and Demiriz, A. 2000. Constrained k-means clustering. *Microsoft Research, Redmond* 1–8.

Cai, D.; He, X.; and Han, J. 2005. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering* 17(12):1624–1637.

Chang, X.; Nie, F.; Ma, Z.; and Yang, Y. 2014. Balanced k-means and min-cut clustering. *Computer Science*.

Chen, W.-Y.; Song, Y.; Bai, H.; Lin, C.-J.; and Chang, E. Y. 2011. Parallel spectral clustering in distributed systems. *Pattern Analysis and Machine Intelligence (PAMI), IEEE Transactions On* 33(3):568–586.

Chen, L. 2014. Dual linear regression based classification for face cluster recognition. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2673–2680. IEEE.

Cheng, G.; Han, J.; Guo, L.; Liu, Z.; Bu, S.; and Ren, J. 2015. Effective and efficient midlevel visual elements-oriented land-use classification using VHR remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing* 53(8):4238–4249.

Du, Z.; Liu, Y.; and Qian, D. 2009. An energy-efficient balanced clustering algorithm for wireless sensor networks. In *The 5th International Conference on Wireless Communications, Networking and Mobile Computing*, 1–4. IEEE.

Eckstein, J., and Bertsekas, D. P. 1992. On the douglas rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming* 55(1-3):293–318.

Han, J.; Zhang, D.; Hu, X.; Guo, L.; Ren, J.; and Wu, F. 2015. Background prior-based salient object detection via deep reconstruction residual. *IEEE Transactions on Circuits and Systems for Video Technology* 25(8):1309–1321.

Lu, X.; Wu, H.; and Yuan, Y. 2014. Double constrained nmf for hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing* 52(5):2746–2758.

Lu, X.; Yuan, Y.; and Yan, P. 2013. Image super-resolution via double sparsity regularized manifold learning. *IEEE Transactions on Circuits and Systems for Video Technology* 23(12):2022–2033.

Malinen, M. I., and Fränti, P. 2014. Balanced k-means for clustering. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, 32–41. Springer.

Ng, A. Y.; Jordan, M. I.; Weiss, Y.; et al. 2002. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems (Proceedings of NIPS)*, volume 2, 849–856. MIT; 1998.

Nie, F.; Xu, D.; Tsang, I. W.; and Zhang, C. 2009. Spectral embedded clustering. In *IJCAI*, 1181–1186.

Nie, F.; Zeng, Z.; Tsang, I. W.; Xu, D.; and Zhang, C. 2011. Spectral embedded clustering: a framework for in-sample and out-of-sample spectral clustering. *IEEE Transactions on Neural Networks* 22(11):1796–808.

Nie, F.; Wang, X.; and Huang, H. 2014. Clustering and projected clustering with adaptive neighbors. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 977–986.

Nie, F.; Wang, X.; and Huang, H. 2016. The constrained laplacian rank algorithm for graph-based clustering. In *AAAI Conference on Artificial Intelligence*.

Powell, M. J. D. 1969. A method for nonlinear constraints in minimization problems. *Optimization* 5(6):283–298.

Strehl, A., and Ghosh, J. 2002. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research* 3(Dec):583–617.

Tahir, M. A.; Chan, C.-H.; Kittler, J.; and Bouridane, A. 2011. Face recognition using multi-scale local phase quantisation and linear regression classifier. In *Proceedings of the 18th IEEE International Conference on Image Processing*, 765–768. IEEE.

Trevor, H.; Robert, T.; and Jerome, F. 2001. The elements of statistical learning: Data mining, inference and prediction. *New York: Springer-Verlag* 1(8):371–406.

Ye, J.; Zhao, Z.; and Wu, M. 2008. Discriminative k-means for clustering. In *Advances in Neural Information Processing Systems (Proceedings of NIPS)*, 1649–1656.

Ye, J. 2007. Least squares linear discriminant analysis. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, 1087–1093. ACM.

Zhong, S., and Ghosh, J. 2003. Model-based clustering with soft balancing. In *The 3rd IEEE International Conference on Data Mining (ICDM)*, 459–466. IEEE.

Zou, H.; Hastie, T.; and Tibshirani, R. 2006. Sparse principal component analysis. *Journal of Computational and Graphical Statistics* 15(2):265–286.