Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Reduction Techniques for Graph-Based Convex Clustering

Lei Han[1] **and Yu Zhang**[2*]
[1]Department of Statistics, Rutgers University
[2]Department of Computer Science and Engineering, Hong Kong University of Science and Technology
[1]lhan@stat.rutgers.edu, leihan.cs@gmail.com; [2]yu.zhang.ust@gmail.com

Abstract

The Graph-based Convex Clustering (GCC) method has gained increasing attention recently. The GCC method adopts a fused regularizer to learn the cluster centers and obtains a geometric clusterpath by varying the regularization parameter. One major limitation is that solving the GCC model is computationally expensive. In this paper, we develop efficient graph reduction techniques for the GCC model to eliminate edges, each of which corresponds to two data points from the same cluster, without solving the optimization problem in the GCC method, leading to improved computational efficiency. Specifically, two reduction techniques are proposed according to tree-based and cyclic-graph-based convex clustering methods separately. The proposed reduction techniques are appealing since they only need to scan the data once with negligibly additional cost and they are independent of solvers for the GCC method, making them capable of improving the efficiency of any existing solver. Experiments on both synthetic and real-world datasets show that our methods can largely improve the efficiency of the GCC model.

Introduction

Clustering, which partitions a data set into several subsets with each one having similar data points, is an important unsupervised task, and it has been extensively studied in the literature. Many clustering algorithms have been proposed such as the k-means method, density-based approaches (Ester et al. 1996), spectral clustering methods (Ng, Jordan, and Weiss 2002), and minimum spanning tree (MST) algorithms (Grygorash, Zhou, and Jorgensen 2006; Wang, Wang, and Wilkes 2009). Different algorithms have their own favors to capture certain types of cluster structure. For example, the k-means algorithm can group data points with each cluster having a convex shape but the spectral clustering and MST methods can deal with the non-convex shape and even more complex one. All the aforementioned clustering methods have some limitations, e.g., some of them relying on the initial setting of the cluster center since it is difficult to attain the global optimum and many of them failing to determine the number of clusters.

Recently, the GCC method (Hocking et al. 2011; Chen et al. 2014) has attracted increasing attentions since it can determine the number of clusters based on the globally optimal solution of its convex objective function. The GCC model organizes the data points as an undirected graph and adopts a fused regularizer as the sum of a set of pairwise fusion terms each of which corresponds to an edge in the graph. Then, the GCC method obtains a clusterpath by optimizing the objective functions according to a sequence of regularization parameters. The generated clusterpath provides a meaningfully geometric interpretation for the clustering structure hidden in the data. For the clustering performance, the GCC method is comparable to the state-of-the-art clustering methods such as the spectral clustering. However, solving the GCC model is very computationally expensive due to the complex fused regularizer.

In this paper, we develop novel graph reduction techniques for the GCC model. Instead of seeking efficient algorithms to solve the GCC model, the graph reduction techniques are developed to eliminate edges in the graph before optimizing the objective function in the GCC model such that the data points connected by the eliminated edges are (almost) guaranteed to be grouped. In this way, we can reduce the number of variable to be optimized since the variables for the data points to be detected are from the same cluster and will be merged as one variable, hence we can improve the computational efficiency. Those reduction techniques are obtained from the analysis on the relations between the primal and dual forms of the GCC model. The proposed reduction techniques have good match with the clusterpath generated by the GCC method since the reduction techniques can identity the data points to be grouped together when varying the value of the regularization parameter, which is just what the clusterpath needs. The proposed reduction techniques are appealing since they only need to scan the data once with negligibly additional cost and they can improve the efficiency of any existing solver for the GCC model since they are independent of solvers. Specifically, we consider two types of graphs in the GCC model, i.e., the tree which is acyclic and the generally cyclic graph, and correspondingly two reduction techniques are proposed for the tree-based convex clustering (TCC) model and the cyclic-graph-based convex clustering (CGCC) model, respectively. We evaluate the proposed reduction techniques

*Both authors contributed equally.

on both synthetic and real-world datasets and find that they can gain considerable speedup.

Overview on The GCC Method

If unspecified, we use bold-face and capital letters for matrices, bold-face and lower-case letters for vectors, and lower-case letters for scalars. We are given n data points in a p-dimensional space and the data matrix is denoted by $\mathbf{X} \in \mathbb{R}^{n\times p}$, where $\mathbf{x}_i$, the ith row of $\mathbf{X}$, denotes the ith data point. There is an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to encode relations between the n data points, where each data point corresponds to a vertex in $\mathcal{V}$ and $\mathcal{E}$ denotes the set of edges in $\mathcal{G}$. The GCC method proposed in (Hocking et al. 2011) aims to solve the following convex optimization problem:

$$\min_{\mathbf{A}\in\mathbb{R}^{n\times p}} \frac{1}{2}\|\mathbf{A}-\mathbf{X}\|_F^2 + \lambda \sum_{(i,j)\in\mathcal{E}} w_{ij}\|\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_j\|_q, \qquad (1)$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm, $\boldsymbol{\alpha}_i$ is the ith row of $\mathbf{A}$, $w_{ij} \geq 0$ denotes the similarity between the corresponding pair of data points, and $\|\cdot\|_q$ denotes the ℓ_q norm of a vector. Here $\boldsymbol{\alpha}_i$ can be viewed as an instance-specific cluster center for $\mathbf{x}_i$ and different data points having the same instance-specific cluster center will be considered to belong to the same cluster. So the fused regularizer (i.e., the second term in problem (1)) is to detect the equivalence among all the possible pairs of rows in the matrix $\mathbf{A}$ based on $\mathcal{E}$. According to (Chen et al. 2014), the graph $\mathcal{G}$ is only required to be connected and hence it can be sparse where many w_{ij}'s are equal to zero or equivalently the set of edges $\mathcal{E}$ has a small size. The continuous path of the optimal solutions obtained from problem (1) by varying λ is called the *clusterpath* (Hocking et al. 2011; Chen et al. 2014).

In the following analysis, we consider two instantiations of the GCC model depending on the structure of $\mathcal{G}$, i.e. the TCC and CGCC models. We first consider the simple case where $\mathcal{G}$ is a tree and correspondingly problem (1) defines the TCC model, and then study a general graph, which can be cyclic, with problem (1) as the objective function of the CGCC model.

Tree Reduction for The TCC Model

In this section, we consider the TCC model. The tree used is denoted by $\mathcal{T} = (\mathcal{V}, \mathcal{E}_\mathcal{T})$. It is obvious that $|\mathcal{E}_\mathcal{T}|$, the size of $\mathcal{E}_\mathcal{T}$, is equal to $n-1$.

The Dual Form for The TCC Model

We first transform problem (1) for the TCC model. The fused regularizer, denoted by $\Omega_q(\mathbf{A})$, in problem (1) can be reformulated as $\Omega_q(\mathbf{A}) = \lambda\|\mathbf{WCA}\|_{1,q}$, where $\|\cdot\|_{1,q}$ denotes the sum of the ℓ_q norms of the rows in a matrix, $\mathbf{W} \in \mathbb{R}^{(n-1)\times(n-1)}$ is a diagonal matrix with the weights w_{ij}'s in the diagonal for $(i,j) \in \mathcal{E}_\mathcal{T}$, and $\mathbf{C} \in \mathbb{R}^{(n-1)\times n}$ is an auxiliary sparse matrix with each row containing only two non-zero entries 1 and -1 corresponding to an edge in $\mathcal{E}_\mathcal{T}$. Therefore, the objective function of the TCC model can be reformulated as

$$\min_{\mathbf{A}\in\mathbb{R}^{n\times p}} \frac{1}{2}\|\mathbf{A}-\mathbf{X}\|_F^2 + \lambda\|\mathbf{WCA}\|_{1,q}. \qquad (2)$$

Now, we propose some useful lemmas.

Lemma 1. *$\mathbf{C}$ has full row-rank and so $rank(\mathbf{C}) = n-1$.*

Lemma 2. *By constructing a matrix $\mathbf{D} = \begin{bmatrix} \mathbf{C} \\ \mathbf{1}_n \end{bmatrix} \in \mathbb{R}^{n\times n}$, where $\mathbf{1}_n \in \mathbb{R}^{1\times n}$ is a row vector with all elements being 1, then $rank(\mathbf{D}) = n$ and hence $\mathbf{D}$ is invertible.*

Based on Lemma 1 and 2, let

$$\begin{bmatrix} \mathbf{B} \\ \boldsymbol{\eta} \end{bmatrix} = \boldsymbol{\Gamma} = \mathbf{DA} = \begin{bmatrix} \mathbf{CA} \\ \mathbf{1}_n\mathbf{A} \end{bmatrix} \in \mathbb{R}^{n\times p}, \qquad (3)$$

where $\mathbf{B} \in \mathbb{R}^{(n-1)\times p}$ and $\boldsymbol{\eta} \in \mathbb{R}^{1\times p}$. Then problem (2) can be rewritten as $\min_{\boldsymbol{\Gamma}} \frac{1}{2}\|\mathbf{X} - \mathbf{D}^{-1}\boldsymbol{\Gamma}\|_F^2 + \lambda\|\mathbf{WB}\|_{1,q}$. By defining $\mathbf{D}_1 \in \mathbb{R}^{n\times(n-1)}$ and $\mathbf{d}_2 \in \mathbb{R}^{n\times 1}$ as $[\mathbf{D}_1\ \mathbf{d}_2] = \mathbf{D}^{-1}$, we can proceed as

$$\min_{\mathbf{B},\boldsymbol{\eta}} f(\mathbf{B},\boldsymbol{\eta}) = \frac{1}{2}\|\mathbf{X} - (\mathbf{D}_1\mathbf{B} + \mathbf{d}_2\boldsymbol{\eta})\|_F^2 + \lambda\|\mathbf{WB}\|_{1,q}.$$

Now by setting $\frac{\partial f}{\partial \boldsymbol{\eta}} = 0$, the solution of $\boldsymbol{\eta}$ is given by

$$\boldsymbol{\eta}^* = \left(\mathbf{d}_2^T\mathbf{d}_2\right)^{-1}\mathbf{d}_2^T(\mathbf{X} - \mathbf{D}_1\mathbf{B}^*). \qquad (4)$$

Note that $\mathbf{d}_2^T\mathbf{d}_2$ is a scalar. By plugging Eq. (4) back into $f(\mathbf{B},\boldsymbol{\eta})$, we can rewrite problem (2) as

$$\min_{\mathbf{B}\in\mathbb{R}^{(n-1)\times p}} \frac{1}{2}\|\widetilde{\mathbf{X}} - \widetilde{\mathbf{D}}\mathbf{B}\|_F^2 + \lambda\|\mathbf{WB}\|_{1,q}, \qquad (5)$$

where $\mathbf{I}$ denotes an identity matrix with appropriate size, $\widetilde{\mathbf{X}} = \left(\mathbf{I} - \frac{1}{\mathbf{d}_2^T\mathbf{d}_2}\mathbf{d}_2\mathbf{d}_2^T\right)\mathbf{X} \in \mathbb{R}^{n\times p}$, and $\widetilde{\mathbf{D}} = \left(\mathbf{I} - \frac{1}{\mathbf{d}_2^T\mathbf{d}_2}\mathbf{d}_2\mathbf{d}_2^T\right)\mathbf{D}_1 \in \mathbb{R}^{n\times(n-1)}$. The solution of problem (5) has an affine relationship with the solution of problem (2), where $\mathbf{A}^* = \mathbf{D}^{-1}\boldsymbol{\Gamma}^*$ and $\boldsymbol{\Gamma}^*$ can be obtained from $\mathbf{B}^*$ based on Eqs. (3) and (4). In order to find the sufficient condition that guarantees some data points to be grouped into a cluster, we derive the dual form of problem (5) as[1]

$$\begin{aligned} \min_{\boldsymbol{\Theta}\in\mathbb{R}^{n\times p}} \quad & g(\boldsymbol{\Theta}) = \frac{\lambda^2}{2}\left\|\boldsymbol{\Theta} - \frac{\widetilde{\mathbf{X}}}{\lambda}\right\|_F^2 - \|\widetilde{\mathbf{X}}\|_F^2, \\ \text{s.t.} \quad & \left\|\widetilde{\mathbf{d}}_r\boldsymbol{\Theta}\right\|_{\bar{q}} \leq w_r,\ r = 1,\cdots,n-1, \end{aligned} \qquad (6)$$

where $\boldsymbol{\Theta} \in \mathbb{R}^{n\times p}$ is the dual variable, the index r corresponds to a pair of indices (i,j) denoted by $r \leftrightarrow (i,j)$, w_r is equal to w_{ij}, $\widetilde{\mathbf{d}}_r$ is the rth row of $\widetilde{\mathbf{D}}^T$, and $\bar{q} = \frac{q}{q-1}$. The KKT conditions for problem (5) or (6) include

$$\begin{aligned} &\lambda\boldsymbol{\Theta}^* = \widetilde{\mathbf{X}} - \widetilde{\mathbf{D}}\mathbf{B}^*, \\ &\widetilde{\mathbf{d}}_r\boldsymbol{\Theta}^* = \begin{cases} \frac{\boldsymbol{\beta}_r^*}{\|\boldsymbol{\beta}_r^*\|_{\bar{q}}}, & \text{if } \boldsymbol{\beta}_r^* \neq \mathbf{0}, (\text{i.e., } \boldsymbol{\alpha}_i \neq \boldsymbol{\alpha}_j), \\ \mathbf{u} \text{ with } \|\mathbf{u}\|_{\bar{q}} \leq w_i, & \text{if } \boldsymbol{\beta}_r^* = \mathbf{0}, (\text{i.e., } \boldsymbol{\alpha}_i = \boldsymbol{\alpha}_j), \end{cases} \end{aligned} \qquad (7)$$

where $r \leftrightarrow (i,j)$ and $\boldsymbol{\beta}_r^*$ is the rth row of $\mathbf{B}^*$. Eq. (7) suggests a sufficient condition to determine whether two data points will be clustered by the TCC method as

$$\left\|\widetilde{\mathbf{d}}_r\boldsymbol{\Theta}^*\right\|_{\bar{q}} < w_r \Rightarrow \boldsymbol{\beta}_r^* = \mathbf{0} \Leftrightarrow \boldsymbol{\alpha}_i = \boldsymbol{\alpha}_j. \qquad (R1)$$

$$\boldsymbol{\alpha}_i = \boldsymbol{\alpha}_j,\ \boldsymbol{\alpha}_j = \boldsymbol{\alpha}_k \Rightarrow \boldsymbol{\alpha}_i = \boldsymbol{\alpha}_k. \qquad (R2)$$

[1] Please refer to the supplementary material (http://www.stat.rutgers.edu/home/lhan/) for the details.

According to rules (R1) and (R2), if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are grouped by the TCC method which is equivalent to $\boldsymbol{\beta}_r^* = 0$, the corresponding edge in the tree is useless and can be eliminated since we will treat $\boldsymbol{\alpha}_i$ and $\boldsymbol{\alpha}_i$ as the same variable. Unfortunately, using rule (R1) is not feasible since we do not know the optimal solution $\boldsymbol{\Theta}^*$ and neither does rule (R2).

Inspired by the feature screening methods (Wang et al. 2013; Wang, Wonka, and Ye 2014; Wang et al. 2014; Wang and Ye 2014; Zhao and Liu 2014), we resort to constructing a feasible region $\boldsymbol{\mathcal{O}}$ which contains $\boldsymbol{\Theta}^*$, and then relax rule (R1) as

$$\sup_{\boldsymbol{\Theta}} \left\{ \left\| \widetilde{\mathbf{d}}_r \boldsymbol{\Theta} \right\|_{\bar{q}} : \boldsymbol{\Theta} \in \boldsymbol{\mathcal{O}} \right\} < w_r \Rightarrow \boldsymbol{\beta}_r^* = \mathbf{0}. \tag{8}$$

Eq. (8) provides a sufficient condition to find the edges to be eliminated. The key here is to construct a tight region $\boldsymbol{\mathcal{O}}$ for $\boldsymbol{\Theta}^*$, since a tighter region $\boldsymbol{\mathcal{O}}$ concentrating around $\boldsymbol{\Theta}^*$ will lead to the elimination of more edges.

Feasible Region Construction

If we are given the optimal $\boldsymbol{\Theta}^*(\lambda')$ at some regularization parameter λ' where the optimal solution $\boldsymbol{\Theta}^*$ is viewed as a function of the regularization parameter, then we can construct $\boldsymbol{\mathcal{O}}$ for $\boldsymbol{\Theta}^*(\lambda)$, where $\lambda < \lambda'$, by utilizing $\boldsymbol{\Theta}^*(\lambda')$ as we will see later. This motivates the reduction techniques to work along a decreasing sequence of regularization parameters, through which the clusterpath can be generated simultaneously.

We first determine a constant $\lambda_{\max}$ such that for any regularization parameter λ larger than $\lambda_{\max}$, the TCC model will group all the data points into one cluster. $\lambda_{\max}$ can be used for the first regularization parameter since the corresponding optimal solution can be obtained analytically without any solver. For problem (6), we define $\boldsymbol{\mathcal{F}}_r = \{\boldsymbol{\Theta} : \|\widetilde{\mathbf{d}}_r \boldsymbol{\Theta}\|_{\bar{q}} \leq w_r\}$ for $r = 1, \cdots, n-1$ and $\boldsymbol{\mathcal{F}} = \cap_{r=1,\cdots,n-1} \boldsymbol{\mathcal{F}}_r$ as the intersection of $\{\boldsymbol{\mathcal{F}}_r\}_{r=1}^{n-1}$. It is easy to see that if $\widetilde{\mathbf{X}}/\lambda \in \boldsymbol{\mathcal{F}}$, then $\boldsymbol{\Theta}^*(\lambda) = \widetilde{\mathbf{X}}/\lambda$. Moreover, from (R1) we can see that if $\widetilde{\mathbf{X}}/\lambda$ is an interior point of $\boldsymbol{\mathcal{F}}$ then $\mathbf{B}^* = \mathbf{0}$. Indeed, we have the following result to determine $\lambda_{\max}$.

Theorem 1. *For problem (6), we define* $\lambda_{\max} = \max_r \{\rho_r : \|\frac{\widetilde{\mathbf{d}}_r \widetilde{\mathbf{X}}}{\rho_r}\|_{\bar{q}} = w_r\}$. *Then the following statements are equivalent: (i)* $\frac{\widetilde{\mathbf{X}}}{\lambda} \in \boldsymbol{\mathcal{F}}$; *(ii)* $\boldsymbol{\Theta}^*(\lambda) = \frac{\widetilde{\mathbf{X}}}{\lambda}$; *(iii)* $\mathbf{B}^* = \mathbf{0}$.

The condition (iii) in Theorem 1 implies that all the data points are grouped together to a cluster for a regularization parameter $\lambda_{\max}$. Now, we can construct $\boldsymbol{\mathcal{O}}$ for any λ smaller than $\lambda_{\max}$ via the following theorem.[2]

Theorem 2. *Suppose that the optimal* $\boldsymbol{\Theta}^*(\lambda')$ *is known for a* $\lambda' \leq \lambda_{\max}$. *Let* ρ_r *be defined in Theorem (1). For any* $0 < \lambda < \lambda'$, *define* $\mathbf{d}_* = \arg\max_{\widetilde{\mathbf{d}}_r} \rho_r$, $\mathbf{e}_* = \mathbb{I}\left(\left| \frac{\mathbf{d}_* \widetilde{\mathbf{X}}}{\lambda_{\max}} \right| - \left\| \frac{\mathbf{d}_* \widetilde{\mathbf{X}}}{\lambda_{\max}} \right\|_\infty \doteq 0 \right)$,

$$\mathbf{n}(\lambda') = \begin{cases} \frac{\widetilde{\mathbf{X}}}{\lambda'} - \boldsymbol{\Theta}^*(\lambda'), & \text{if } \lambda' < \lambda_{\max}, \\ \mathbf{d}_*^T \operatorname{sign}\left(\frac{\mathbf{d}_* \widetilde{\mathbf{X}}}{\lambda_{\max}} \right), & \text{if } \lambda' = \lambda_{\max},\ \bar{q} = 1, \\ \mathbf{d}_*^T \frac{\mathbf{d}_* \widetilde{\mathbf{X}}}{\lambda_{\max}}, & \text{if } \lambda' = \lambda_{\max},\ \bar{q} = 2, \\ \mathbf{e}_*^T \frac{\mathbf{d}_* \widetilde{\mathbf{X}}}{\lambda_{\max}}, & \text{if } \lambda' = \lambda_{\max},\ \bar{q} = \infty, \end{cases}$$

and $\mathbf{v}(\lambda, \lambda') = \frac{\widetilde{\mathbf{X}}}{\lambda} - \boldsymbol{\Theta}^*(\lambda')$, $\mathbf{v}^\perp(\lambda, \lambda') = \mathbf{v}(\lambda, \lambda') - \frac{\langle \mathbf{v}(\lambda,\lambda'), \mathbf{n}(\lambda') \rangle}{\|\mathbf{n}(\lambda')\|_F^2} \mathbf{n}(\lambda')$, *where* $\mathbb{I}(\cdot)$, $|\cdot|$, *and* $\doteq$ *are element-wise indicator, absolute, and equivalent operators, and* $\operatorname{sign}(\cdot)$ *denotes the sign function. Then, we have the following result:*

$$\left\| \boldsymbol{\Theta}^*(\lambda) - \mathbf{o}(\lambda, \lambda') \right\|_F \leq R(\lambda, \lambda'), \tag{9}$$

where $\mathbf{o}(\lambda, \lambda') = \boldsymbol{\Theta}^*(\lambda') + \frac{1}{2}\mathbf{v}^\perp(\lambda, \lambda')$ *and* $R(\lambda, \lambda') = \frac{1}{2}\left\| \mathbf{v}^\perp(\lambda, \lambda') \right\|_F$.

Theorem 2 implies that the optimal $\boldsymbol{\Theta}^*(\lambda)$ lies in a ball depending on $\boldsymbol{\Theta}^*(\lambda')$ with the center $\mathbf{o}(\lambda, \lambda')$ and radius $R(\lambda, \lambda')$. Therefore, we can directly use the ball to construct the feasible region $\boldsymbol{\mathcal{O}}(\lambda, \lambda')$:

$$\boldsymbol{\mathcal{O}}(\lambda, \lambda') = \left\{ \boldsymbol{\Theta}(\lambda) : \left\| \boldsymbol{\Theta}(\lambda) - \mathbf{o}(\lambda, \lambda') \right\|_F \leq R(\lambda, \lambda') \right\}. \tag{10}$$

Based on the sufficient condition in Eq. (8), we only need to estimate the following supreme value:

$$\sup_{\boldsymbol{\Theta}} \left\{ \left\| \widetilde{\mathbf{d}}_r \boldsymbol{\Theta} \right\|_{\bar{q}} : \boldsymbol{\Theta} \in \boldsymbol{\mathcal{O}}(\lambda, \lambda'),\ r = 1, \cdots, n-1 \right\}. \tag{11}$$

The Eater Rules

The supreme value can be obtained in the following theorem which also concludes the Exact Tree Reduction (Eater) rules finally.

Theorem 3. *(Eater Rules) For the TCC model, suppose the optimal solution* $\boldsymbol{\Theta}^*(\lambda')$ *at* $0 < \lambda' \leq \lambda_{\max}$ *is known. Let* $r \leftrightarrow (i, j)$. *Then for* $0 < \lambda < \lambda'$, *the edge* (i, j) *can be eliminated under a regularization parameter* λ, *i.e.,* $\boldsymbol{\alpha}_i^* = \boldsymbol{\alpha}_j^*$, *if the following conditions hold:*

$$\begin{cases} \left\| \widetilde{\mathbf{d}}_r \mathbf{o}(\lambda, \lambda') \right\|_\infty + R(\lambda, \lambda') \|\widetilde{\mathbf{d}}_r\|_2 < w_r, & \text{if } q = 1, \\ \left\| \widetilde{\mathbf{d}}_r \mathbf{o}(\lambda, \lambda') \right\|_2 + R(\lambda, \lambda') \|\widetilde{\mathbf{d}}_r\|_2 < w_r, & \text{if } q = 2, \\ \left\| \widetilde{\mathbf{d}}_r \mathbf{o}(\lambda, \lambda') \right\|_1 + R(\lambda, \lambda') \|\widetilde{\mathbf{d}}_r\|_2 < w_r, & \text{if } q = \infty, \end{cases} \tag{R1*}$$

$$\boldsymbol{\alpha}_i^* = \boldsymbol{\alpha}_k^* \text{ and } \boldsymbol{\alpha}_j^* = \boldsymbol{\alpha}_k^* \text{ for some } k. \tag{R2*}$$

Some remarks for the Eater rules: *(1) In the Eater rules, all the conditions can be determined efficiently via only simple matrix operations on the given data matrix* $\mathbf{X}$. *This is promising since we can directly determine which data points will be clustered together for a given* λ *without solving the objective function. The only requirement is the knowledge of* $\boldsymbol{\Theta}^*(\lambda')$ *at some* $\lambda' > \lambda$. *(2) In order to generate the clusterpath, multiple TCC models need to be learned along a decreasing sequence of regularization parameters* $\lambda_0 > \lambda_1 > \cdots > \lambda_t$ *where* $\lambda_0 = \lambda_{\max}$. *The proposed reduction techniques can work based on this sequence, since* $\boldsymbol{\Theta}^*(\lambda_i)$ *can be used to do the reduction for the regularization parameter* λ_{i+1}. *Moreover, at the beginning, both* $\lambda_{\max}$ *and* $\boldsymbol{\Theta}^*(\lambda_{\max})$ *can be analytically computed according to Theorem 1.*

[2] Similar to (Hocking et al. 2011), three values for q (i.e., 1, 2, and ∞) are investigated and our analysis can be easily generalized to a general q.

Graph Reduction for CGCC

In this section, we present reduction techniques for the CGCC model. The cyclic graph is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}_{\mathcal{C}})$ and $\overline{\mathbf{C}} \in \mathbb{R}^{m\times n}$ is defined similar to $\mathbf{C}$, where m, the number of edges, is not smaller than n. Then we have the following properties for $\overline{\mathbf{C}}$.

Lemma 3. *$\overline{\mathbf{C}} \in \mathbb{R}^{m\times n}$ is a rank-deficient matrix, where $rank(\overline{\mathbf{C}}) < n \leq m$.*

Since $\overline{\mathbf{C}}$ is rank-deficient, we cannot transform the objective function of the CGCC model in a similar way to the TCC model, and therefore we investigate the original problem (1) instead.

The Dual Form for The CGCC Model

Problem (1) can be reformulated as

$$\min_{\mathbf{A}\in\mathbb{R}^{n\times p}} \frac{1}{2}\|\mathbf{A}-\mathbf{X}\|_F^2 + \lambda\|\overline{\mathbf{W}}\overline{\mathbf{C}}\mathbf{A}\|_{1,q}, \tag{12}$$

where $\overline{\mathbf{W}}$ is the corresponding diagonal weight matrix. The Lagrangian dual for problem (12) is

$$\begin{aligned} \min_{\mathbf{\Phi}\in\mathbb{R}^{m\times p}} \quad & h(\mathbf{\Phi}) = \frac{\lambda^2}{2}\left\|\overline{\mathbf{C}}^T\mathbf{\Phi} - \frac{\mathbf{X}}{\lambda}\right\|_F^2 - \|\mathbf{X}\|_F^2, \\ \text{s.t.} \quad & \|\boldsymbol{\phi}_r\|_{\bar{q}} \leq \overline{w}_r,\ r = 1,\cdots,m, \end{aligned} \tag{13}$$

where $\boldsymbol{\phi}_r$ is a row of $\mathbf{\Phi}$. The KKT conditions for problem (13) include

$$\mathbf{X} = \mathbf{A}^* + \lambda\overline{\mathbf{C}}^T\mathbf{\Phi}^*,\ \overline{\mathbf{C}}\mathbf{X} = \overline{\mathbf{C}}\mathbf{A}^* + \lambda\overline{\mathbf{C}}\overline{\mathbf{C}}^T\mathbf{\Phi}^*, \tag{14}$$

$$\boldsymbol{\phi}_r^* = \begin{cases} \frac{[\overline{\mathbf{C}}\mathbf{A}^*]_r}{\|[\overline{\mathbf{C}}\mathbf{A}^*]_r\|_{\bar{q}}}, & \text{if } [\overline{\mathbf{C}}\mathbf{A}^*]_r \neq \mathbf{0} \text{ (i.e., } \boldsymbol{\alpha}_i \neq \boldsymbol{\alpha}_j), \\ \mathbf{u} \text{ with } \|\mathbf{u}\|_{\bar{q}} \leq \overline{w}_r, & \text{if } [\overline{\mathbf{C}}\mathbf{A}^*]_r = \mathbf{0} \text{ (i.e., } \boldsymbol{\alpha}_i = \boldsymbol{\alpha}_j), \end{cases} \tag{15}$$

where $r \leftrightarrow (i,j)$. According to Eq. (15), we obtain a sufficient condition as

$$\|\boldsymbol{\phi}_r^*\|_{\bar{q}} < \overline{w}_r \Rightarrow [\overline{\mathbf{C}}\mathbf{A}^*]_r = \mathbf{0} \Leftrightarrow \boldsymbol{\alpha}_i = \boldsymbol{\alpha}_j. \tag{R3}$$

Similar to TCC, we want to construct a feasible region $\boldsymbol{\mathcal{G}}$ containing $\mathbf{\Phi}^*$, and then relax rule (R3) as

$$\sup_{\mathbf{\Phi}} \{\|\boldsymbol{\phi}_r\|_{\bar{q}} : \mathbf{\Phi} \in \boldsymbol{\mathcal{G}}\} < \overline{w}_r \Rightarrow [\overline{\mathbf{C}}\mathbf{A}^*]_r = \mathbf{0}. \tag{16}$$

In order to construct $\boldsymbol{\mathcal{G}}$, similar to the TCC model, we first need to seek a constant $\overline{\lambda}_{\max}$ under which the CGCC model can group all the data points into a cluster. We define $\overline{\boldsymbol{\mathcal{F}}}$ as $\overline{\boldsymbol{\mathcal{F}}} = \{\mathbf{\Phi} : \|\boldsymbol{\phi}_r\|_{\bar{q}} < \overline{w}_r, r = 1,\cdots,m\}$. For $\mathbf{\Phi} \in \overline{\boldsymbol{\mathcal{F}}}$, the optimality condition of problem (13) implies that

$$\overline{\mathbf{C}}\overline{\mathbf{C}}^T\mathbf{\Phi}^* = \frac{\overline{\mathbf{C}}\mathbf{X}}{\lambda}. \tag{17}$$

It is easy to prove that $\overline{\mathbf{C}}\overline{\mathbf{C}}^T$ is rank-deficient and it is not invertible since $\overline{\mathbf{C}} \in \mathbb{R}^{m\times n}$ is rank-deficient. As a consequence, problem (13) can have multiple solutions and so does Eq. (17), bringing difficulties to find $\overline{\lambda}_{\max}$. Instead of deriving exact reduction rules for the CGCC model, we propose to seek for inexact reduction rules, which is called inexact Cyclic-Graph Reduction (Cigar) rules.

The Cigar Rules

We propose to relax the dual problem (13) as

$$\begin{aligned} \min_{\overline{\mathbf{\Phi}}} \quad & \overline{h}(\overline{\mathbf{\Phi}}) = \frac{\lambda^2}{2}\left\|\overline{\mathbf{C}}^T\overline{\mathbf{\Phi}} - \frac{\mathbf{X}}{\lambda}\right\|_F^2 - \|\mathbf{X}\|_F^2 + \frac{\delta\lambda^2}{2}\|\overline{\mathbf{\Phi}}\|_F^2, \\ \text{s.t.} \quad & \|\overline{\boldsymbol{\phi}}_r\|_{\bar{q}} \leq \overline{w}_r,\ r = 1,\cdots,m, \end{aligned} \tag{18}$$

where δ is a small positive constant. Problem (18) is strictly convex. $\overline{\mathbf{D}}$ is defined as $\overline{\mathbf{D}} = \overline{\mathbf{C}}\overline{\mathbf{C}}^T + \delta\mathbf{I} \in \mathbb{R}^{m\times m}$ and is positive definite. Then we can determine $\overline{\lambda}_{\max}$ in the following theorem.

Theorem 4. *For the relaxed dual problem (18), we define $\overline{\lambda}_{\max} = \max_r\{\rho_r : \|\frac{\overline{\mathbf{d}}_r\overline{\mathbf{Y}}}{\rho_r}\|_{\bar{q}} = \overline{w}_r\}$, where $\overline{\mathbf{d}}_r$ denotes the rth row of the inverse matrix $\overline{\mathbf{D}}^{-1}$ and $\overline{\mathbf{Y}} = \overline{\mathbf{C}}\mathbf{X}$. Then the following statements are equivalent: (i) $\frac{\overline{\mathbf{D}}^{-1}\overline{\mathbf{Y}}}{\lambda} \in \overline{\boldsymbol{\mathcal{F}}}$; (ii) $\overline{\mathbf{\Phi}}^*(\lambda) = \frac{\overline{\mathbf{D}}^{-1}\overline{\mathbf{Y}}}{\lambda}$.*

Then we can construct the feasible region for $\overline{\mathbf{\Phi}}^*$ based on the following theorem.

Theorem 5. *Suppose that $\overline{\mathbf{\Phi}}^*(\lambda')$ is known for $\lambda' < \overline{\lambda}_{\max}$. For any $0 < \lambda < \lambda'$, we define $\overline{\mathbf{n}}(\lambda') = \frac{\mathbf{\Lambda}^{-1}\overline{\mathbf{Y}}}{\lambda'} - \mathbf{\Lambda}\overline{\mathbf{\Phi}}^*(\lambda')$, $\overline{\mathbf{v}}(\lambda,\lambda') = \frac{\mathbf{\Lambda}^{-1}\overline{\mathbf{Y}}}{\lambda} - \mathbf{\Lambda}\overline{\mathbf{\Phi}}^*(\lambda')$, $\overline{\mathbf{v}}^{\perp}(\lambda,\lambda') = \overline{\mathbf{v}}(\lambda,\lambda') - \frac{\langle\overline{\mathbf{v}}(\lambda,\lambda'),\overline{\mathbf{n}}(\lambda')\rangle}{\|\overline{\mathbf{n}}(\lambda')\|_F^2}\overline{\mathbf{n}}(\lambda')$. Then, we have the following result:*

$$\left\|\mathbf{\Lambda}\overline{\mathbf{\Phi}}^*(\lambda) - \overline{\mathbf{o}}(\lambda,\lambda')\right\|_F \leq \overline{R}(\lambda,\lambda'),\ \textit{if}\ \lambda' < \overline{\lambda}_{\max}, \tag{19}$$

where $\mathbf{\Lambda} = \overline{\mathbf{D}}^{\frac{1}{2}}$, $\overline{\mathbf{o}}(\lambda,\lambda') = \mathbf{\Lambda}\overline{\mathbf{\Phi}}^*(\lambda') + \frac{1}{2}\overline{\mathbf{v}}^{\perp}(\lambda,\lambda')$, and $\overline{R}(\lambda,\lambda') = \frac{1}{2}\left\|\overline{\mathbf{v}}^{\perp}(\lambda,\lambda')\right\|_F$.

Finally, we obtain the Cigar rules for the CGCC model.

Definition 1. *(Cigar Rules) For the CGCC problem, suppose the optimal solution $\overline{\mathbf{\Phi}}^*(\lambda')$ of problem (18) is given where $0 < \lambda' < \lambda_{\max}$. Let $r \leftrightarrow (i,j)$ and let $\boldsymbol{\zeta}_r$ be the rth row of $\mathbf{\Lambda}^{-1}$. Then for $0 < \lambda < \lambda'$, the edge (i,j) can be eliminated under λ, i.e. $\mathbf{x}_i$ and $\mathbf{x}_j$ clustered (inexactly), which is denoted by $\alpha_i^* \simeq \alpha_j^*$, if the following conditions hold:*

$$\begin{cases} \|\boldsymbol{\zeta}_r\mathbf{o}(\lambda,\lambda')\|_\infty + R(\lambda,\lambda')\|\boldsymbol{\zeta}_r\|_2 < w_r, & \textit{if } q = 1, \\ \|\boldsymbol{\zeta}_r\mathbf{o}(\lambda,\lambda')\|_2 + R(\lambda,\lambda')\|\boldsymbol{\zeta}_r\|_2 < w_r, & \textit{if } q = 2, \\ \|\boldsymbol{\zeta}_r\mathbf{o}(\lambda,\lambda')\|_1 + R(\lambda,\lambda')\|\boldsymbol{\zeta}_r\|_2 < w_r, & \textit{if } q = \infty, \end{cases} \tag{R3*}$$

$$\boldsymbol{\alpha}_i^* \simeq \boldsymbol{\alpha}_k^* \text{ and } \boldsymbol{\alpha}_j^* \simeq \boldsymbol{\alpha}_k^* \textit{ for some } k. \tag{R4*}$$

Some remarks for the Cigar rules: *(1) The Cigar rules are inexact, since problem (18) is not the dual form of problem (1). When we use the Cigar rules, some edges may be mis-eliminated and as a consequence, some data points may be mis-clustered. (2) The parameter δ in problem (18) plays an important role, since $\overline{\lambda}_{\max}$ depends on the value of δ. Actually, we can determine δ empirically and we put the details in the supplementary material. (3) When the number of rows in $\overline{\mathbf{C}}$, i.e. m, is large, directly calculating $\overline{\mathbf{D}}^{-1}$, $\mathbf{\Lambda}$, and $\mathbf{\Lambda}^{-1}$ is computationally demanding. However, due to the specific form of $\overline{\mathbf{D}}$, there exists efficient ways to compute those matrices and we put the details in the supplementary material.*

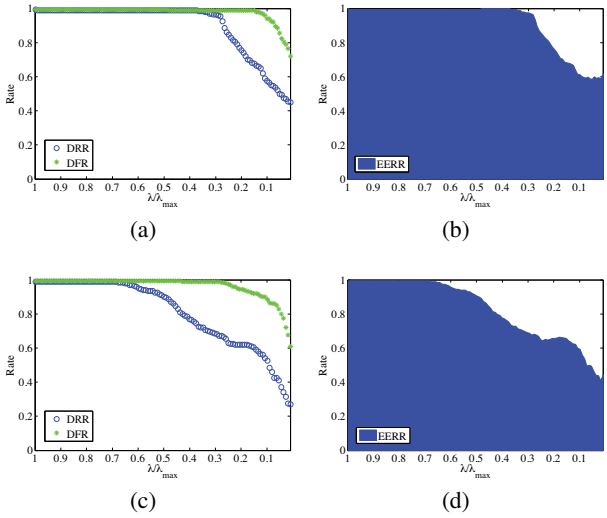

Figure 1: The performance of the Eater rule on synthetic data (n=200). The first and second rows report the results in the Halfmoon and Spiral data respectively.

Experiments

In this section, we evaluate the Eater and Cigar rules on both synthetic and real-world datasets. For the TCC model, we use the the tree structure generated by the MST algorithm, since the MST clustering methods can deal with complex cluster structure. For the CGCC model, we add edges to the MST to construct cyclic graphs, in which the Cigar rule is evaluated. In order to measure the performance of the reduction rules, we define the following metrics: *data reduction rate* (DRR) = $1-\widehat{n}_c/n$, *data fusion rate* (DFR) = $1-n_c^*/n$, *exact edge reduction rate* (EERR) = $|\widehat{\mathbf{E}}_{re}|/|\mathbf{E}_{re}^*|$, *mistaken edge-elimination rate* (MEER)=$|\widehat{\mathbf{E}}_{re}\cap(\mathbf{E}-\mathbf{E}_{re}^*)|/|\widehat{\mathbf{E}}_{re}|$, where $\widehat{n}_c$ is the number of distinct $\boldsymbol{\alpha}_i$'s (number of clusters) obtained by the Eater/Cigar rule, n_c^* is the true number of distinct $\boldsymbol{\alpha}_i$'s detected by some solver, $\widehat{\mathbf{E}}_{re}$ is the set of removed edges by Eater/Cigar rules, and $\mathbf{E}_{re}^*$ is the true set of removed edges. The MEER is specially defined for the Cigar rule since the Eater rule is exact.

Similar to (Hocking et al. 2011), the weight before any data pair (i,j) is defined as $w_{ij}=\exp\left(-\gamma\|\mathbf{x}_i-\mathbf{x}_j\|_2^2\right)$, and we use $\gamma=10/\bar{d}^2$, where $\bar{d}$ is the average ℓ_2 distance for all possible pairs of data points. In the experiments, q is set to be 2. We use the FISTA (Beck and Teboulle 2009) algorithm to solve problem (5) for the TCC model. For the CGCC model, we use the ADMM method to solve it. All the experiments are conducted on a machine with Intel i7 CPU and 8GB RAM under the Matlab 2013b environment.

Experiments on Synthetic Data

We investigate two widely used synthetic datasets in the clustering literature (Grygorash, Zhou, and Jorgensen 2006; Wang, Wang, and Wilkes 2009), i.e. the halfmoon and spiral datasets. In each dataset, we generate n data points where n is equal to 200. The Eater and Cigar rules are tested on a sequence of the regularization parameter with 500 values for λ equally spaced between $\lambda_{\max}$ and $\lambda_{\max}/500$ and the clusterpath can be generated simultaneously. In the following experiments, we use CGCC-k to denote the CGCC model on a cyclic graph generated from the MST by adding $k(n-1)$ additional edges with largest weights. In our experiments, we find that all the TCC and CGCC models can correctly detect the clusters in the data, and the clustering performance of these models is reported in the supplementary material. In the following, we focus on evaluating the proposed rules.

Table 1: Running Time (in seconds) on the synthetic data based on the Eater rule. E+S denotes the total time cost of using the Eater rule and the solver.

Data	Solver	Eater	E+S	Speedup
halfmoon (n=200)	134.4	0.5	21.6	**6.2**
spiral (n=200)	176.4	0.6	31.3	**5.6**

Fig. 1 shows the performance of the Eater rule in terms of DRR, DFR and EERR when n equals 200. In Figs. 1(a) and 1(c), the closer the DRR curve is to the DFR curve, the better performance the Eater rule gains, because the region below the DRR curve denotes the fraction of data reduction that the Eater rule can achieve. Note that if the rule is exact, the DRR curve will never exceed the DFR curve. By comparing the DRR and DFR curves, we can see that a large proportion of the data points are clustered via the Eater rule. The blue regions in Figs. 1(b) and 1(d) represent the exact edge reduction rate. We can see that at least over 40% edges can be eliminated for each $\lambda/\lambda_{\max}$, and the average fraction of the eliminated edges is around 70%. Those results verify that the Eater rule is able to detect the data points from the same cluster without learning the TCC model. Table 1 shows the running time for generating the clusterpath by using the TCC model, where the use of the Eater rule can achieve 5-6 times speedup compared with solving the TCC problem directly.

Fig. 2 shows the performance of the Cigar rule in terms of DRR, DFR, EERR and MEER, when the size of the data is 200. Different from the Eater rule, the Cigar rule may miscluster some data points as analyzed previously. The red regions in Figs. 2(c), (f), (i) and (l) denote the fraction of mistaken eliminated edges under different settings. From the results, we observe that some mistakes are made by the Cigar rule when λ is small. Moreover, the Cigar rule becomes less effective in terms of the EERR when λ is small. On the contrast, when $\lambda/\lambda_{\max}>0.3$, the Cigar rule can correctly detect the clusters. Based on the DFR curve, we see that when λ is small, the data fusion rate varies rapidly, implying that the clusterpath changes frequently. This is a possible reason that the Cigar rule becomes less effective for small λ. Table 2 records the running time of different methods and we find that the Cigar rule can achieve 2-7 times speedup.

Iris Data & Vehicle Data

We test the Eater and Cigar rules on two UCI datasets, i.e., the iris and vehicle datasets.[3] The iris data contains 150 data

[3] https://archive.ics.uci.edu/ml/datasets.html

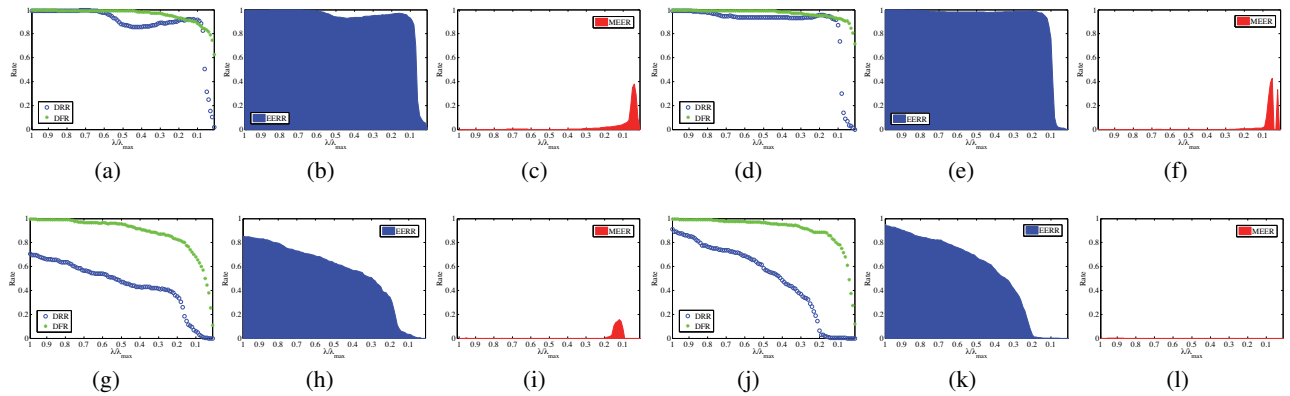

Figure 2: The performance of the Cigar rule on synthetic data (n=200). The first and second rows denote the results on the halfmoon and spiral datasets respectively. In each row, the first three figures denote the results for the CGCC-1 model while the rest is for the CGCC-2 model.

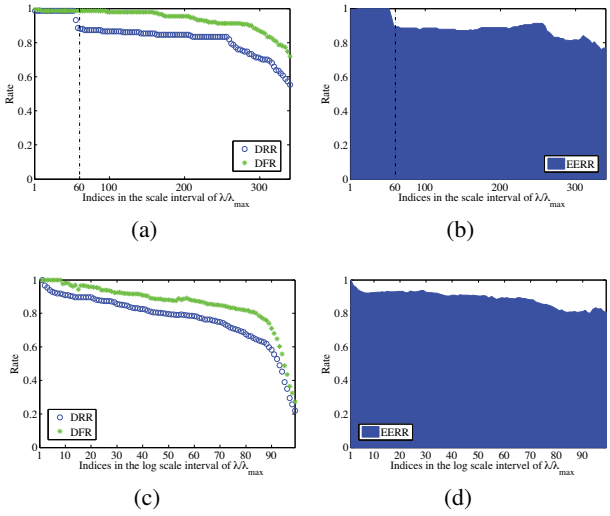

Figure 3: The performance of the Eater rule on the iris and vehicle data. The first and second rows report the results in the Iris and Vehicle data respectively.

points and each data point has 4 attributes, while the vehicle data contains 946 data points, each of which has 18 attributes. In the iris data, we first test the Eater rule along a sequence of 360 values. The sequence consists of two parts, where the first subsequence contains 60 values equally spaced on the logarithmic scale from $\frac{1}{1.1}$ to $\frac{1}{1.1^{60}}$ and the second one contains 300 values equally spaced on the residual interval $(\frac{1}{1.1^{60}}, 0)$, since empirically we find that the clusterpath changes frequently when $\lambda/\lambda_{\max} < \frac{1}{1.1^{60}}$. For the CGCC model, we generate a sequence of λ identical to that in the synthetic data. In the vehicle data, we test the two rules along a sequence of 100 values equally spaced on the logarithmic scale of $\lambda/\lambda_{\max}$ from $\frac{1}{1.1}$ to $\frac{1}{1.1^{100}}$.

Fig. 3 shows the performance of the Eater rule. In Figs. 3(b) and 3(d), we observe that the Eater rule can consistently achieve 80%-90% EERR, resulting in very effective data reduction as shown in Figs. 3(a) and 3(c). Fig. 4 shows the performance of the Cigar rule for the CGCC model. In Figs. 4(b), (e), (h) and (k), although the Cigar rule only eliminates a small fraction of the edges when $\lambda/\lambda_{\max} < 0.1$, it is almost exact as shown in Figs. 4(c), (f), (i) and (l), implying that the Cigar rule is more effective in the real-world datasets than the synthetic datasets. Table 3 shows the running time of different methods in the two datasets. From the results, we can see that the proposed rules can achieve around 2-9 times speedup. Moreover, similar to the results on the synthetic data, the Eater rule is much more efficient than the Cigar rule, but the computation cost for both the rules can be negligible compared with the time cost of the solver.

Table 2: Running Time (in seconds) on the synthetic data based on the Cigar rule. C+S means the total time cost of using the Cigar rule and the solver.

Data	Model	Solver	Cigar	C+S	Speedup
Halfmoon (n=200)	CGCC-1	189.3	8.1	25.0	**7.6**
	CGCC-2	201.6	15.1	42.9	**4.7**
Spiral (n=200)	CGCC-1	202.2	6.9	85.2	**2.4**
	CGCC-2	233.4	13.9	88.2	**2.6**

Table 3: Running time (seconds) on the real-world datasets. R+S means the total time cost of using the reduction rules and the solver.

Data	Model	Solver	Rule	R+S	Speedup
Iris	TCC	247.6	0.3	28.5	**8.7**
	CGCC-1	310.9	7.1	45.4	**6.8**
	CGCC-2	388.1	12.8	58.7	**6.6**
Vehicle	TCC	1399.5	2.6	168.7	**8.3**
	CGCC-1	2625.7	31.4	877.5	**3.0**
	CGCC-2	3473.7	44.4	1271.4	**2.7**

Conclusion and Future Work

In this paper, we developed novel reduction techniques for the graph-based convex clustering model to eliminated a fraction of edges in the graph before solving it. In the future

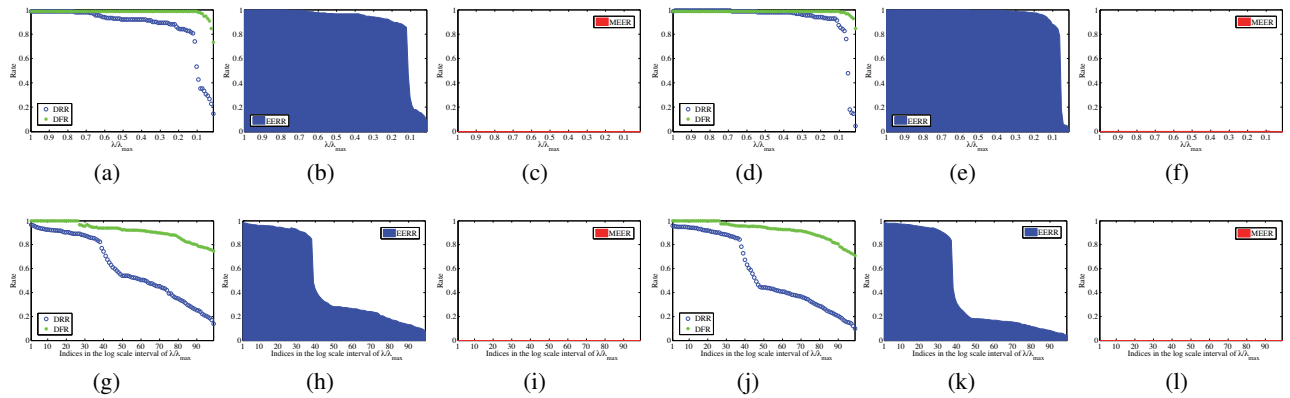

Figure 4: The performance of the Cigar rule on the iris and vehicle data. The first and second rows denote the performance in the iris and vehicle data respectively. In each row, the first three figures denote the results for the CGCC-1 model while the rest is for the CGCC-2 model.

study, we are interested in finding exact rules for the CGCC model and applying the reduction techniques to other applications with large-scale data.

Acknowledgment

This work is supported by NSFC (61305071, 61473087) and Nature Science Foundation of Jiangsu Provice of China (BK20141340).

References

Beck, A., and Teboulle, M. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* 2(1):183–202.

Chen, G. K.; Chi, E.; Ranola, J.; and Lange, K. 2014. Convex clustering: An attractive alternative to hierarchical clustering. *arXiv preprint arXiv:1409.2065*.

Ester, M.; Kriegel, H.-P.; Sander, J.; and Xu, X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, volume 96, 226–231.

Grygorash, O.; Zhou, Y.; and Jorgensen, Z. 2006. Minimum spanning tree based clustering algorithms. In *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence*, 73–81.

Hocking, T. D.; Joulin, A.; Bach, F.; and Vert, J.-P. 2011. Clusterpath: An algorithm for clustering using convex fusion penalties. In *Proceedings of the 28th International Conference on Machine Learning*.

Ng, A. Y.; Jordan, M. I.; and Weiss, Y. 2002. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, 849–856.

Wang, J., and Ye, J. 2014. Two-layer feature reduction for sparse-group lasso via decomposition of convex sets. In *Advances in Neural Information Processing Systems*, 2132–2140.

Wang, J.; Zhou, J.; Wonka, P.; and Ye, J. 2013. Lasso screening rules via dual polytope projection. In *Advances in Neural Information Processing Systems*, 1070–1078.

Wang, J.; Zhou, J.; Wonka, P.; and Ye, J. 2014. A safe screening rule for sparse logistic regression. In *Advances in Neural Information Processing Systems*.

Wang, X.; Wang, X.; and Wilkes, D. M. 2009. A divide-and-conquer approach for minimum spanning tree-based clustering. *IEEE Transactions on Knowledge and Data Engineering* 21(7):945–958.

Wang, J.; Wonka, P.; and Ye, J. 2014. Scaling SVM and least absolute deviations via exact data reduction. In *Proceedings of International Conference on Machine Learning*.

Zhao, Z., and Liu, J. 2014. Safe and efficient screening for sparse support vector machine. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.