Constrained 1-Spectral Clustering

Syama Sundar Rangapuram
srangapu@mpi-inf.mpg.de
Max Planck Institute for Computer Science
Saarland University, Saarbrücken, Germany

Matthias Hein
hein@cs.uni-saarland.de
Saarland University, Saarbrücken
Germany

Abstract

An important form of prior information in clustering comes in form of cannot-link and must-link constraints. We present a generalization of the popular spectral clustering technique which integrates such constraints. Motivated by the recently proposed 1-spectral clustering for the unconstrained problem, our method is based on a tight relaxation of the constrained normalized cut into a continuous optimization problem. Opposite to all other methods which have been suggested for constrained spectral clustering, we can always guarantee to satisfy all constraints. Moreover, our soft formulation allows to optimize a trade-off between normalized cut and the number of violated constraints. An efficient implementation is provided which scales to large datasets. We outperform consistently all other proposed methods in the experiments.

1 Introduction

The task of clustering is to find a natural grouping of items given e.g. pairwise similarities. In real world problems, such a natural grouping is often hard to discover with given similarities alone or there is more than one way to group the given items. In either case, clustering methods benefit from domain knowledge that gives bias to the desired clustering. Wagstaff et. al (Wagstaff et al., 2001) are the first to consider constrained clustering by encoding available domain knowledge in the form of pairwise must-link (ML, for short) and cannot-link (CL) constraints. By incorporating these constraints into k-means they achieve much better performance. Since acquiring such constraint information is relatively easy, constrained clustering has become an active area of research; see (Basu et al., 2008) for an overview.

Spectral clustering is a graph-based clustering algorithm originally derived as a relaxation of the NP-hard normalized cut problem. The spectral relaxation leads to an eigenproblem for the graph Laplacian, see (Hagen & Kahng, 1991; Shi & Malik, 2000; von Luxburg, 2007). However, it is known that the spectral relaxation can be quite loose (Guattery & Miller, 1998). More recently, it has been shown that one can equivalently rewrite the discrete (combinatorial) normalized Cheeger cut problem into a continuous optimization problem using the nonlinear 1-graph Laplacian (Hein & Bühler, 2010; Szlam & Bresson, 2010) which yields much better cuts than the spectral relaxation. In further work it is shown that even all balanced graph cut problems, including normalized cut, have a tight relaxation into a continuous optimization problem (Hein & Setzer, 2011).

The first approach to integrate constraints into spectral clustering was based on the idea of modifying the weight matrix in order to enforce the must-link and cannot-link constraints and then solving the resulting unconstrained problem (Kamvar et al., 2003). Another idea is to adapt the embedding obtained from the first k eigenvectors of the graph Laplacian (Li et al., 2009). Closer to the original normalized graph cut problem are the approaches that start with the optimization problem of the spectral relaxation and add constraints that encode must-links and cannot-links (Yu & Shi, 2004; Eriksson et al., 2007; Xu et al., 2009; Wang & Davidson, 2010). Furthermore, the case where the constraints are allowed to be inconsistent is considered in (Coleman et al., 2008).

In this paper we contribute in various ways to the area of graph-based constrained learning. First, we show in the spirit of 1-spectral clustering (Hein & Bühler, 2010; Hein & Setzer, 2011), that the *constrained* normalized cut problem has a *tight* relaxation as an *unconstrained*

Appearing in Proceedings of the 15^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2012, La Palma, Canary Islands. Volume 22 of JMLR: W&CP 22.

continuous optimization problem. Our method, which we call COSC, is the first one in the field of constrained spectral clustering, which can guarantee that all given constraints are fulfilled. While we present arguments that in practice it is the best choice to satisfy all constraints even if the data is noisy, in the case of inconsistent or unreliable constraints one should refrain from doing so. Thus our second contribution is to show that our framework can be extended to handle degree-of-belief and even inconsistent constraints. In this case COSC optimizes a trade-off between having small normalized cut and a small number of violated constraints. We present an efficient implementation of COSC based on an optimization technique proposed in (Hein & Setzer, 2011) which scales to large datasets. While the continuous optimization problem is non-convex and thus convergence to the global optimum is not guaranteed, we can show that our method improves any given partition which satisfies all constraints or it stops after one iteration.

All omitted proofs and additional experimental results can be found in the supplementary material.

Notation. Set functions are denoted by a hat, $\hat{S}$, while the corresponding extension is S. In this paper, we consider the normalized cut problem with general vertex weights. Formally, let $G(V, E, w, b)$ be an undirected graph G with vertex set V and edge set E together with edge weights $w : V \times V \to \mathbb{R}_+$ and vertex weights $b : V \to \mathbb{R}_+$ and $n = |V|$. Let $C \subset V$ and denote by $\overline{C} = V \backslash C$. We define respectively the cut, the generalized volume and the normalized cut (with general vertex weights) of a partition $(C, \overline{C})$ as

$$\mathrm{cut}(C, \overline{C}) = 2 \sum_{i \in C,\, j \in \overline{C}} w_{ij}, \qquad \mathrm{gvol}(C) = \sum_{i \in C} b_i,$$

$$\mathrm{bal}(C) = 2\frac{\mathrm{gvol}(C)\mathrm{gvol}(\overline{C})}{\mathrm{gvol}(V)}, \quad \mathrm{NCut}(C, \overline{C}) = \frac{\mathrm{cut}(C, \overline{C})}{\mathrm{bal}(C)}.$$

We obtain ratio cut and normalized cut for special cases of the vertex weights, $b_i = 1$, and $b_i = d_i$, where $d_i = \sum_{j=1}^n w_{ij}$, respectively. In the ratio cut case, $\mathrm{gvol}(C)$ is the cardinality of C and in the normalized cut case, it is volume of C, denoted by $\mathrm{vol}(C)$.

2 The Constrained Normalized Cut Problem

We consider the normalized cut problem with must-link and cannot-link constraints. Let $G(V, E, w, b)$ denote the given graph and Q^m, Q^c be the constraint matrices, where the element q^m_{ij} (or q^c_{ij}) $\in \{0, 1\}$ specifies the must-link (or cannot-link) constraint between i and j. We will in the following always assume that G is connected. All what is stated below and our suggested algorithm can be easily generalized to *degree of belief constraints* by allowing q^m_{ij} (and q^c_{ij}) $\in [0, 1]$. However, in the following we consider only q_{ij} (and q^c_{ij}) $\in \{0, 1\}$, in order to keep the theoretical statements more accessible.

Definition 2.1. *We call a partition $(C, \overline{C})$ **consistent** if it satisfies all constraints in Q^m and Q^c.*

Then the **constrained normalized cut problem** is to minimize $\mathrm{NCut}(C, \overline{C})$ over all consistent partitions. If the constraints are unreliable or inconsistent one can relax this problem and optimize a trade-off between normalized cut and the number of violated constraints. In this paper, we address both problems in a common framework.

We define the set functions, $M, N : 2^V \to \mathbb{R}$, as

$$\hat{M}(C) := 2 \sum_{i \in C,\, j \in \overline{C}} q^m_{ij}$$

$$\hat{N}(C) := \sum_{i \in C,\, j \in C} q^c_{ij} + \sum_{i \in \overline{C},\, j \in \overline{C}} q^c_{ij} = \mathrm{vol}(Q^c) - 2 \sum_{i \in C,\, j \in \overline{C}} q^c_{ij}.$$

$\hat{M}(C)$ and $\hat{N}(C)$ are equal to twice the number of violated must-link and cannot-link constraints of partition $(C, \overline{C})$.

As we show in the following, both the constrained normalized cut problem and its soft version can be addressed by minimization of $\hat{F}_\gamma : 2^V \to \mathbb{R}$ defined as

$$\hat{F}_\gamma(C) = \frac{\mathrm{cut}(C, \overline{C}) + \gamma \left(\hat{M}(C) + \hat{N}(C) \right)}{\mathrm{bal}(C)}, \qquad (1)$$

where $\gamma \in \mathbb{R}_+$. Note that $\hat{F}_\gamma(C) = \mathrm{NCut}(C, \overline{C})$ if $(C, \overline{C})$ is consistent. Thus the minimization of $\hat{F}_\gamma(C)$ corresponds to a trade-off between having small normalized cut and satisfying all constraints parameterized by γ.

The relation between the parameter γ and the number of violated constraints by the partition minimizing $\hat{F}_\gamma$ is quantified in the following lemma.

Lemma 2.1. *Let $(C, \overline{C})$ be consistent and $\lambda = \mathrm{NCut}(C, \overline{C})$. If $\gamma \geq \frac{\mathrm{gvol}(V)}{4(l+1)}\lambda$, then any minimizer of $\hat{F}_\gamma$ violates no more than l constraints.*

Note that it is easy to construct a partition which is consistent and thus the above choice of γ is constructive. The following theorem is immediate from the above lemma for the special case $l = 0$.

Theorem 2.1. *Let $(C, \overline{C})$ be consistent with the given constraints and $\lambda = \mathrm{NCut}(C, \overline{C})$. Then for $\gamma \geq \frac{\mathrm{gvol}(V)}{4}\lambda$, it holds that*

$$\operatorname*{arg\,min}_{\substack{C \subset V: \\ (C, \overline{C})\ consistent}} \mathrm{NCut}(C, \overline{C}) = \operatorname*{arg\,min}_{C \subset V} \hat{F}_\gamma(C)$$

and the optimum values of both problems are equal.

Thus the constrained normalized cut problem can be equivalently formulated as the combinatorial problem of minimizing $\hat{F}_\gamma$. In the next section we will show that this problem allows for a tight relaxation into a continuous optimization problem.

2.1 A tight continuous relaxation of $\hat{F}_\gamma$

Minimizing $\hat{F}_\gamma$ is a hard combinatorial problem. In the following, we derive an equivalent continuous optimization problem. Let $f : \mathbb{R}^V \to \mathbb{R}$ denote a function on V, and $\mathbf{1}_C$ denote the vector that is 1 on C and 0 elsewhere. Define

$$M(f) := \sum_{i,j=1}^{n} q^m_{ij} \, |f_i - f_j| \text{ and}$$

$$N(f) := \mathrm{vol}(Q^c)\big(\max(f) - \min(f)\big) - \sum_{i,j=1}^{n} q^c_{ij} \, |f_i - f_j| \,,$$

where $\max(f)$ and $\min(f)$ are respectively the maximum and minimum elements of f. Note that $M(\mathbf{1}_C) = \hat{M}(C)$ and $N(\mathbf{1}_C) = \hat{N}(C)$ for any non-trivial[1] partition $(C, \overline{C})$.

Let B denote the diagonal matrix with the vertex weights b on the diagonal. We define

$$F_\gamma(f) = \frac{\sum_{i,j=1}^{n} w_{ij} \, |f_i - f_j| + \gamma \; M(f) + \gamma \; N(f)}{\left\| B(f - \frac{1}{\mathrm{gvol}(V)} \, \langle f, b \rangle \, \mathbf{1}) \right\|_1}.$$

We denote the numerator of $F_\gamma(f)$ by $R_\gamma(f)$ and the denominator by $S(f)$.

Lemma 2.2. *For any non-trivial partition it holds that $\hat{F}_\gamma(C) = F_\gamma(\mathbf{1}_C)$.*

From Lemma 2.2 it immediately follows that minimizing F_γ is a relaxation of minimizing $\hat{F}_\gamma$. In our main result (Theorem 2.2), we establish that both problems are actually equivalent, so that we have a tight relaxation. In particular a minimizer of F_γ is an indicator function corresponding to the optimal partition minimizing $\hat{F}_\gamma$.

The proof is based on the following key property of the functional F_γ. Given any non-constant $f \in \mathbb{R}^n$, optimal thresholding,

$$C^*_f = \underset{\min_i f_i \, \leq \, t \, < \, \max_i f_i}{\arg\min} \hat{F}_\gamma(C^t_f),$$

where $C^t_f = \{i \in V | f_i > t\}$, yields an indicator function on some $C^*_f \subset V$ with smaller or equal value of F_γ.

[1] A partition $(C, \overline{C})$ is non-trivial if neither $C = \emptyset$ nor $C = V$.

Theorem 2.2. *For $\gamma \geq 0$, we have*

$$\min_{C \subset V} \hat{F}_\gamma(C) \;=\; \min_{f \in \mathbb{R}^n, \; f \text{ non-constant}} F_\gamma(f).$$

Moreover, a solution of the first problem can be obtained from the solution of the second problem.

Proof. It has been shown in (Hein & Bühler, 2010), that

$$\sum_{i,j=1}^{n} w_{ij} |f_i - f_j| = \int_{-\infty}^{\infty} \mathrm{cut}(C^t_f, \overline{C^t_f}) dt$$

We define $\hat{P} : 2^V \to \mathbb{R}$ as $\hat{P}(C) = 1$, if $C \neq V$ and $C \neq \emptyset$, and 0 otherwise. Denoting by $\mathrm{cut}_Q(C, \overline{C})$, the cut on the constraint graph whose weight matrix is given by Q, we have

$$\begin{aligned}
R_\gamma(f) = & \int_{-\infty}^{\infty} \mathrm{cut}(C^t_f, \overline{C^t_f}) dt + \gamma \int_{-\infty}^{\infty} \mathrm{cut}_{Q^m}(C^t_f, \overline{C^t_f}) \\
& + \gamma \, \mathrm{vol}(Q^c) \int_{\min_i f_i}^{\max_i f_i} 1 dt - \gamma \int_{-\infty}^{\infty} \mathrm{cut}_{Q^c}(C^t_f, \overline{C^t_f}) \\
= & \int_{-\infty}^{\infty} \mathrm{cut}(C^t_f, \overline{C^t_f}) dt + \gamma \int_{-\infty}^{\infty} \mathrm{cut}_{Q^m}(C^t_f, \overline{C^t_f}) \\
& + \gamma \, \mathrm{vol}(Q^c) \int_{-\infty}^{\infty} \hat{P}(C^t_f) dt - \gamma \int_{-\infty}^{\infty} \mathrm{cut}^c_Q(C^t_f, \overline{C^t_f}) \\
= & \int_{-\infty}^{\infty} \hat{R}_\gamma(C^t_f) \, dt
\end{aligned}$$

Note that $S(f)$ is an even, convex and positively one-homogeneous function.[2] Moreover, every even, convex positively one-homogeneous function, $T : \mathbb{R}^V \to \mathbb{R}$ has the form $T(f) = \sup_{u \in U} \langle u, f \rangle$, where U is a symmetric convex set, see e.g., (Hiriart-Urruty & Lemaréchal, 2001). Note that $S(\mathbf{1}) = 0$ and thus because of the symmetry of U it has to hold $\langle u, \mathbf{1} \rangle = 0$ for all $u \in U$. Since $S(\mathbf{1}_{C^t_f}) = \hat{S}(C^t_f)$ and $\langle u, f \rangle \leq S(f), u \in U$, we have for all $u \in U$,

$$\begin{aligned}
R_\gamma(f) & \geq \int_{-\infty}^{\infty} \frac{\hat{R}_\gamma(C^t_f)}{\hat{S}(C^t_f)} \left\langle u, \mathbf{1}_{C^t_f} \right\rangle dt \\
& \geq \inf_{t \in \mathbb{R}} \frac{\hat{R}_\gamma(C^t_f)}{\hat{S}(C^t_f)} \int_{\min_i f_i}^{\max_i f_i} \left\langle u, \mathbf{1}_{C^t_f} \right\rangle dt, \qquad (2)
\end{aligned}$$

where in the last inequality we changed the limits of integration using the fact that $\langle u, \mathbf{1} \rangle = 0$. Let $C_i := C^t_{f_i}$ and $C_0 = V$. Then

$$\int_{\min_i f_i}^{\max_i f_i} \langle u, \mathbf{1}_{C_i} \rangle \, dt = \sum_{i=1}^{n-1} \langle u, \mathbf{1}_{C_i} \rangle \, (f_{i+1} - f_i) =$$

$$\sum_{i=1}^{n} f_i \big(\langle u, \mathbf{1}_{C_{i-1}} \rangle - \langle u, \mathbf{1}_{C_i} \rangle \big) = \sum_{i=1}^{n} f_i u_i = \; \langle u, f \rangle$$

[2] A function $S : \mathbb{R}^V \to \mathbb{R}$ is positively one-homogeneous if $S(\alpha f) = \alpha \, S(f)$ for all $\alpha > 0$.

Noting that (2) holds for all $u \in U$, we have

$$R_\gamma(f) \geq \inf_{t\in\mathbb{R}} \hat{F}_\gamma(C_f^t) \sup_{u\in U} \langle u, f\rangle = \inf_{t\in\mathbb{R}} \hat{F}_\gamma(C_f^t)\ S(f).$$

This implies that

$$F_\gamma(f) \geq \inf_{t\in\mathbb{R}} \hat{F}_\gamma(C_f^t) = F_\gamma(\mathbf{1}_{C_f^*}), \tag{3}$$

where $C_f^* = \underset{\min_i f_i \,\leq\, t \,<\, \max_i f_i}{\arg\min} \hat{F}_\gamma(C_f^t)$.

This shows that we always get descent by optimal thresholding. Thus the actual minimizer of F_γ is a two-valued function, which can be transformed to an indicator function on some $C \subset V$, because of the scale and shift invariance of F_γ. Then from Lemma 2.2, which shows that for non-trivial partitions, $\hat{F}_\gamma(C) = F_\gamma(\mathbf{1}_C)$, the statement follows. □

Now, we state our second result: the problem of minimizing the functional F_γ over arbitrary real-valued non-constant f, for a particular choice of γ, is in fact equivalent to the NP-hard problem of minimizing normalized cut with constraints. The proof follows directly from Theorem 2.1 and Theorem 2.2.

Theorem 2.3. *Let $(C, \overline{C})$ be consistent and $\lambda = \mathrm{NCut}(C', \overline{C'})$. Then for $\gamma \geq \frac{\mathrm{gvol}(V)}{4}\ \lambda$, it holds that*

$$\min_{\substack{C\subset V:\\ (C,\overline{C})\ consistent}} \mathrm{NCut}(C, \overline{C}) = \min_{f\in\mathbb{R}^n,\ f\ non\text{-}constant} F_\gamma(f)$$

Furthermore, an optimal partition of the constrained problem can be obtained from a minimizer of the right problem.

A few comments on the implications of Theorem 2.3. First, it shows that the constrained normalized cut problem can be equivalently solved by minimizing $F_\gamma(f)$ for the given value of γ. The value of γ depends on the normalized cut value of a partition consistent with given constraints. Note that such a partition can be obtained in polynomial time by 2-coloring the constraint graph as long as the constraints are consistent.

2.2 Integration of must-link constraints via sparsification

If the must-link constraints are reliable and therefore should be enforced, one can directly integrate them by merging the corresponding vertices together with re-definition of edge and vertex weights. In this way ones derives a new reduced graph, where the value of the normalized cut of all partitions that satisfy the must-link constraints are preserved.

The construction of a reduced graph is given below for a must-link constraint (p, q).

1. merge p and q into a single vertex τ.
2. update the vertex weight of τ by $b_\tau = b_p + b_q$.
3. update the edges as follows: if r is any vertex other than p and q, then add an edge between τ and r with weight $w(p, r) + w(q, r)$.

Note that this construction leads to a graph with vertex weights even if the original graph had vertex weights equal to 1. If there are many must-links, one can efficiently integrate all of them together by first constructing the must-link constraint graph and merging each connected component in this way.

The following lemma shows that the above construction preserves all normalized cuts which respect the must-link constraints. We prove it for the simple case where we merge p and q and the proof can easily be extended to the general case by induction.

Lemma 2.3. *Let $G'(V', E', w', b')$ be the reduced graph of $G(V, E, w, b)$ obtained by merging vertices p and q. If a partition $(C, \overline{C})$ does not separate p and q, we have $\mathrm{NCut}_G(C, \overline{C}) = \mathrm{NCut}_{G'}(C', \overline{C'})$.*

All partitions of the reduced graph fulfill all must-link constraints and thus any relaxation of the *unconstrained* normalized cut problem can now be used. Moreover, this is not restricted to the cut criterion we are using but any other graph cut criterion based on cut and the volume of the subsets will be preserved in the reduction.

3 Algorithm for Constrained 1-Spectral Clustering

In this section we discuss the efficient minimization of F_γ based on recent ideas from unconstrained 1-spectral clustering (Hein & Bühler, 2010; Hein & Setzer, 2011). Note, that F_γ is a non-negative ratio of a difference of convex (d.c) function and a convex function, both of which are positively one-homogeneous. In recent work (Hein & Bühler, 2010; Hein & Setzer, 2011), a general scheme, shown in Algorithm 1 (where $\partial S(f)$ denotes the subdifferential of the convex function S at f), is proposed for the minimization of a non-negative ratio of a d.c function and convex function both of which are positively one-homogeneous.

It is shown in (Hein & Setzer, 2011) that Algorithm 1 generates a sequence f^k such that either $F_\gamma(f^{k+1}) < F_\gamma(f^k)$ or the sequence terminates. Moreover, the cluster points of f^k correspond to critical points of F_γ. The scheme is given in Algorithm 1 for the prob-

lem $\min_{f\in\mathbb{R}^n}(R_1(f)-R_2(f))/S(f)$, where

$$R_1(f) := \frac{1}{2}\sum_{i,j=1}^{n}(w_{ij}+\gamma q_{ij}^m)|f_i-f_j| + \frac{\gamma}{2}\sum_{i,j=1}^{n} q_{ij}^c(\max(f)-\min(f))$$

$$R_2(f) := \frac{1}{2}\sum_{i,j=1}^{n} q_{ij}^c|f_i-f_j|,$$

$$S(f) := \frac{1}{2}\left\|B(f-\frac{1}{\mathrm{gvol}(V)}\langle f,b\rangle\,\mathbf{1})\right\|_1$$

Note that R_1, R_2 are both convex functions and $F_\gamma(f)=(R_1(f)-R_2(f))/S(f)$.

Algorithm 1 Minimization of a ratio $(R_1(f)-R_2(f))/S(f)$ where R_1, R_2, S are convex and positively one-homogeneous

1: **Initialization:** f^0 = random with $\|f^0\| = 1$, $\lambda^0 = (R_1(f^0)-R_2(f^0))/S(f^0)$
2: **repeat**
3: $f^{k+1} = \arg\min_{\|f\|_2\le 1}\{R_1(f)-\langle f, r_2\rangle - \lambda^k\langle f,s\rangle\}$ where $r_2\in\partial R_2(f^k)$, $s\in\partial S(f^k)$
4: $\lambda^{k+1} = (R_1(f^{k+1})-R_2(f^{k+1})/S(f^{k+1})$
5: **until** $\frac{|\lambda^{k+1}-\lambda^k|}{\lambda^k} < \epsilon$
6: **Output:** λ^{k+1} and f^{k+1}.

Moreover, it is shown in (Hein & Setzer, 2011), that if one wants to minimize $(R_1(f)-R_2(f))/S(f)$ only over non-constant functions, one has to ensure that $\langle r_2,\mathbf{1}\rangle = \langle s,\mathbf{1}\rangle = 0$. Note, that

$$\partial S(f) = \frac{1}{2}(\mathbb{I}-\frac{1}{\mathrm{gvol}(V)}b\mathbf{1}^T)B\,\mathrm{sign}\left(f-\frac{\langle b,f\rangle}{\mathrm{gvol}(V)}\mathbf{1}\right)$$

$$\partial R_2(f) = \Big\{\sum_{j=1}^{n} q_{ij}^c u_{ij}\,|\,u_{ij}=-u_{ji},\ u_{ij}\in\mathrm{sign}(f_i-f_j)\Big\},$$

where $\mathrm{sign}(x)=[-1,1]$ if $x=0$, otherwise it just the sign function. It is easy to check that $\langle u,\mathbf{1}\rangle = 0$ for all $u\in\partial S(f)$ and all $f\in\mathbb{R}^n$ and there exists always a vector $u\in\partial R_2(f)$ for all $f\in\mathbb{R}^n$ such that $\langle u,\mathbf{1}\rangle = 0$.

In the algorithm the key part is the inner convex problem which one has to solve at each step. In our case it has the form,

$$\min_{\|f\|_2\le 1}\ \frac{1}{2}\sum_{i,j=1}^{n}(w_{ij}+\gamma q_{ij}^m)\,|f_i-f_j| + \frac{\gamma}{2}\sum_{i,j=1}^{n} q_{ij}^c(\max(f)-\min(f)) - \Big\langle f,\gamma\, r_2+\lambda^k s\Big\rangle, \tag{4}$$

where $r_2\in\partial R_2(f^k)$, $s\in\partial S(f^k)$ and $\lambda^k = F_\gamma(f^k)$.

To solve it more efficiently we derive an equivalent smooth dual formulation for this non-smooth convex problem. We replace $w_{ij}+\gamma q_{ij}^m$ by w'_{ij} in the following.

Lemma 3.1. *Let $E\subset V\times V$ denote the set of edges and $A:\mathbb{R}^E\to\mathbb{R}^V$ be defined as $(A\alpha)_i = \sum_{j|(i,j)\in E} w'_{ij}\alpha_{ij}$. Moreover, let U denote the simplex, $U=\{u\in\mathbb{R}^n \mid \sum_{i=1}^n u_i = 1, u_i\ge 0,\ \forall i\}$. The above inner problem is equivalent to*

$$\min_{\left\{\alpha\in\mathbb{R}^E\,|\,\|\alpha\|_\infty\le 1,\alpha_{ij}=-\alpha_{ji}\right\}\atop v\in U}\ \Psi(\alpha,v) := \tag{5}$$
$$c\left\|-A\frac{\alpha}{c}+v+b-P_U\big(-A\frac{\alpha}{c}+v+b\big)\right\|_2,$$

where $c=\frac{\gamma}{2}\mathrm{vol}(Q^c)$, $b=\frac{r_2}{c}+\lambda^k\frac{s}{c}$ and $P_U(x)$ is the projection of x on to the simplex U.

The smooth dual problem can be solved efficiently using first order projected gradient methods like FISTA (Beck & Teboulle, 2009), which has a guaranteed convergence rate of $O(\frac{L}{k^2})$, where k is the number of steps, and L is the Lipschitz constant of the gradient of the objective. The bound on the Lipschitz constant for the gradient of the objective in (5) can be rather loose if the weights are varying a lot. The rescaling of the variable α introduced in Lemma 3.2 leads to a better condition number and also to a tighter bound on the Lipschitz constant. This results in a significant improvement in practical performance.

Lemma 3.2. *Let B be a linear operator defined as $(B\beta)_i := \sum_{j:(i,j)\in E}\beta_{ij}$ and let $s_{ij}=\frac{w'_{ij}}{c}M$, for positive constant $M\ge\|B\|$. The above inner problem is equivalent to*

$$\min_{\{\beta\in\mathbb{R}^E\,|\,\|\beta\|_\infty\le s_{ij},\beta ij=-\beta_{ji}\}\atop v\in U}\ \tilde{\Psi}(\beta,v) := \frac{1}{2}\|d-P_U(d)\|_2^2,$$

where $d=-\frac{B}{M}\beta+v+b$. The Lipschitz constant of the gradient of $\tilde{\Psi}$ is upper bounded by 4.

We can choose M by upper bounding $\|B\|$ using

$$\|B\|^2 \le \max_r \sum_{(r,j)\in E} 1^2 = \max_r \mathrm{neigh}(r),$$

where $\mathrm{neigh}(r)$ is the number of neighbors of vertex r.

Despite the problem of minimizing F_γ is non-convex and thus global convergence is not guaranteed, Algorithm 1 has the following quality guarantee.

Theorem 3.1. *Let $(C,\overline{C})$ be any partition and let $\lambda = \mathrm{NCut}(C,\overline{C})$. If one uses $\mathbf{1}_C$ as the initialization of the Algorithm 1, then the algorithm either terminates in one step or outputs an f^1 which yields a partition $(A,\overline{A})$ such that*

$$\hat{F}_\gamma(A) < \hat{F}_\gamma(C)$$

Moreover, if $(C,\overline{C})$ is consistent and if we set for γ any value larger than $\frac{\mathrm{gvol}(V)}{4}\lambda$ then A is also consistent and $\mathrm{NCut}(A,\overline{A}) < \mathrm{NCut}(C,\overline{C})$.

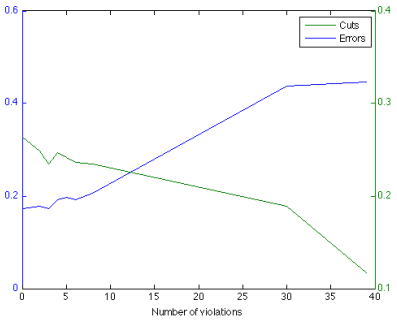


Figure 1: Influence of γ on cut and clustering error.

In practice, the best results can be obtained by first minimizing F_γ for $\gamma = 0$ (unconstrained problem) and then increase the value of γ and use the previously obtained clustering as initialization. This process is iterated until the current partition violates not more than a given number of constraints.

4 Soft- versus Hard-Constrained Normalized Cut Problem

The need for a soft version arises, for example, if the constraints are noisy or inconsistent. Moreover, as we illustrate in the next section, we use the soft version to extend our clustering method to the multi-partitioning problem. Using the bound of Lemma 2.1 for γ, we can solve the soft constrained problem for any given number of violations.

It appears from a theoretical point of view that, due to noise, satisfying all constraints should not be the best choice. However, in our experiments it turned out, that typically the best results were achieved when all constraints were satisfied. We illustrate this behavior for the dataset Sonar, where we generated 80 constraints and increased γ from zero until all constraints were satisfied. In Figure 1, we plot cuts and errors versus the number of violated constraints. One observes that the best error is obtained when all constraints were satisfied. Since by enforcing always all given constraints, our method becomes parameter-free (we increase γ until all constraints are satisfied), we chose this option for the experiments.

5 Multi-Partitioning with Constraints

In this section we present a method to integrate constraints in a multi-partitioning setting. In the multi-partitioning problem, one seeks a k-partitioning $(C_1, \ldots, C_k)$ of the graph such that the normalized multi-cut given by

$$\sum_{i=1}^{k} \mathrm{NCut}(C_i, \overline{C_i}) \tag{6}$$

is minimized. A straightforward way to generate a multi-partitioning is to use a recursive bi-partitioning scheme. Starting with all points as the initial partition, the method repeats the following steps until the current partition has k components.

1. split each of the components in the current partition into two parts.
2. choose among the above splits the one minimizing the multi-cut criterion.

Now we extend this method to the constrained case. Note that it is always possible to perform a binary split which satisfies all must-link constraints. Thus, must-link constraints pose no difficulty in the multi-partitioning scheme, as all must-link constraints can be integrated using the procedure given in 2.2.

However, satisfying all cannot-link constraints is sometimes not possible (cyclic constraints) and usually also not desirable at each level of the recursive bi-partition, since an early binary split cannot separate all classes. The issues here is which cannot-link constraints should be considered for the binary split in step 1.

To address this issue, we use the soft-version of our formulation where we need only to specify the maximum number, l, of violations allowed. We derive this number l assuming the following simple uniform model of the data and constraints. We assume that all classes have equal size and there is an equal number of cannot link constraints between all pairs of classes. Assuming that any binary split does not destroy the class structure, the maximum number of violation is obtained if one class is separated from the rest. Precisely, the expected value of this number, given N cannot-link constraints and k classes, is $\frac{(k-1)(k-2)/2}{k(k-1)/2}\ N$. In the first binary split, these numbers (N and k) are known. In the succesive binary splits, N is known, while k can again be derived, assuming the uniform model, as $\frac{k}{n}\ \tilde{n}$, where $\tilde{n}$ is the size of the current component.

We illustrate our approach using an artificial dataset (mixture of Gaussians, 500 points, 2 dimensions). Figure 2 shows on the left the ground truth and the solution of unconstrained (γ=0) multi-partitioning. In the unconstrained solution, points belonging to the same class are split into two clusters while points from other two classes are merged into a single cluster. On the rightmost, the result of our constrained multi-partitioning framework with 80 randomly generated constraints is shown.

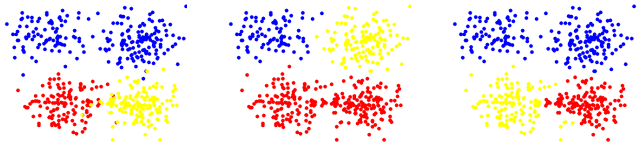


Figure 2: Left: ground-truth, middle: clustering obtained by unconstrained 1-spectral clustering, right: clustering obtained by the constrained version.

Table 1: UCI datasets. The extended MNIST dataset is generated by translating each original input image of MNIST by one pixel, i.e., 8 directions.

Dataset	Size	Features	Classes
Sonar	208	60	2
Spam	4207	57	2
USPS	9298	256	10
MNIST	70000	784	10
MNIST (Ext)	630000	784	10

6 Experiments

We compare our method against the following four related constrained clustering approaches: Spectral Learning (SL) (Kamvar et al., 2003), Flexible Constrained Spectral Clustering (CSP) (Wang & Davidson, 2010), Constrained Clustering via Spectral Regularization (CCSR) (Li et al., 2009) and Spectral Clustering with Linear Constraints (SCLC) (Xu et al., 2009). SL integrates the constraints by simply modifying the weight matrix such that the edges connecting must-links have maximum weight and the edges of cannot-links have zero weight. CSP starts from the spectral relaxation and restricts the space of feasible solutions to those that satisfy a certain amount (specified by the user) of constraints. This amounts to solving a full generalized eigenproblem and choosing among the eigenvectors corresponding to positive eigenvalues the one that has minimum cost. CCSR addresses the problem of incorporating the constraints in the multi-class problem directly by an SDP which aims at adapting the spectral embedding to be consistent with the constraint information. For CSP and CCSR we use the code provided by the authors on their webpages.

In SCLC one solves the spectral relaxation of the normalized cut problem subject to linear constraints (Eriksson et al., 2007; Xu et al., 2009). Cannot-links and must-links are encoded via linear constraints as follows (Eriksson et al., 2007): if the vertices p and q cannot-link (resp. must-link) then add a constraint $f_p = -f_q$ (resp. $f_p = f_q$). Although must-links are correctly formulated, one can argue that the encoding of cannot-links has modeling drawbacks. First observe that any solution that assigns zero to the constrained vertices p and q still satisfies the corresponding cannot-link constraint although it is not feasible to the constrained cut problem. Moreover, one can observe from the derivation of spectral relaxation (von Luxburg, 2007), that vertices belonging to different components need to have only different signs but not the same value. Encoding cannot-links this way introduces bias towards partitions of equal volume, which can be observed in the experiments.

Our evaluation is based on three criteria: clustering error, normalized cut and fraction of constraints violated. For the clustering error we take the known labels and classify each cluster using majority vote. In this way each point is assigned a label and the clustering error is the error of this labeling. We use this measure as it is the expected error one would obtain when using simple semi-supervised learning, where one labels each cluster using majority vote.

The summary of the datasets considered is given in Table 1. The data with missing values are removed and the k-NN similarity graph is constructed from the remaining data as in (Bühler & Hein, 2009). Moreover, redundant data points are removed from the spam dataset. In order to illustrate the performance in case of highly unbalanced problems, we create a binary problem (digit 0 versus rest) from USPS. The constraint pairs are generated in the following manner. We randomly sample pairs of points and for each pair, we introduce a cannot or must-link constraint based on the labels of the sampled pair. The results, averaged over 10 trials are shown in Table 2 for 2-class problems and in Table 3 for multi-class problems[3]. In the plots our method is denoted as COSC and we enforce always all constraints (see discussion in Section 4). Since our formulation is a non-convex problem, we use the best result (based on the achieved cut value) of 10 runs with random initializations. Except our method, no other method can guarantee to satisfy all constraints, even though SCLC does so in all cases. Our method produces always much better cuts than the ones found by SCLC which shows that our method is better suited for solving the constrained normalized cut problem. In terms of the clustering error, our method is consistently better than other methods. In case of unbalanced datasets (Spam, USPS 0 vs rest) our method significantly outperforms SCLC in terms of cuts and clustering error. Moreover, because of hard encoding of constraints, CSLC cannot solve multi-partitioning problems.

[3]CSP could not scale to the large datasets, as the method solves the full (generalized) eigenvalue problem where the matrices involved are not sparse.

Table 2: Results for **binary partitioning**: Left: clustering error versus number of constraints, Middle: normalized cut versus number of constraints, Right: fraction of violated constraints versus number of constraints.

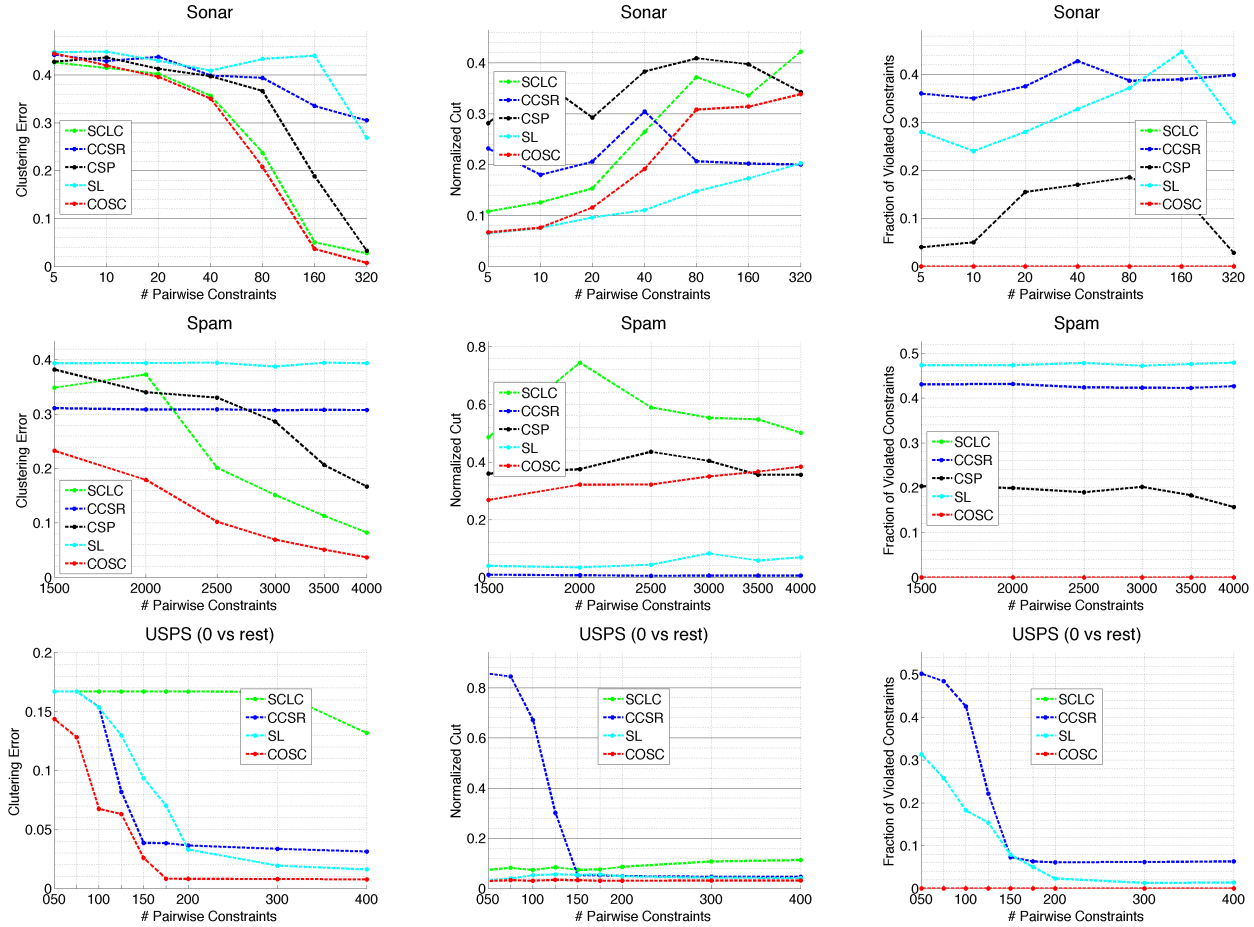
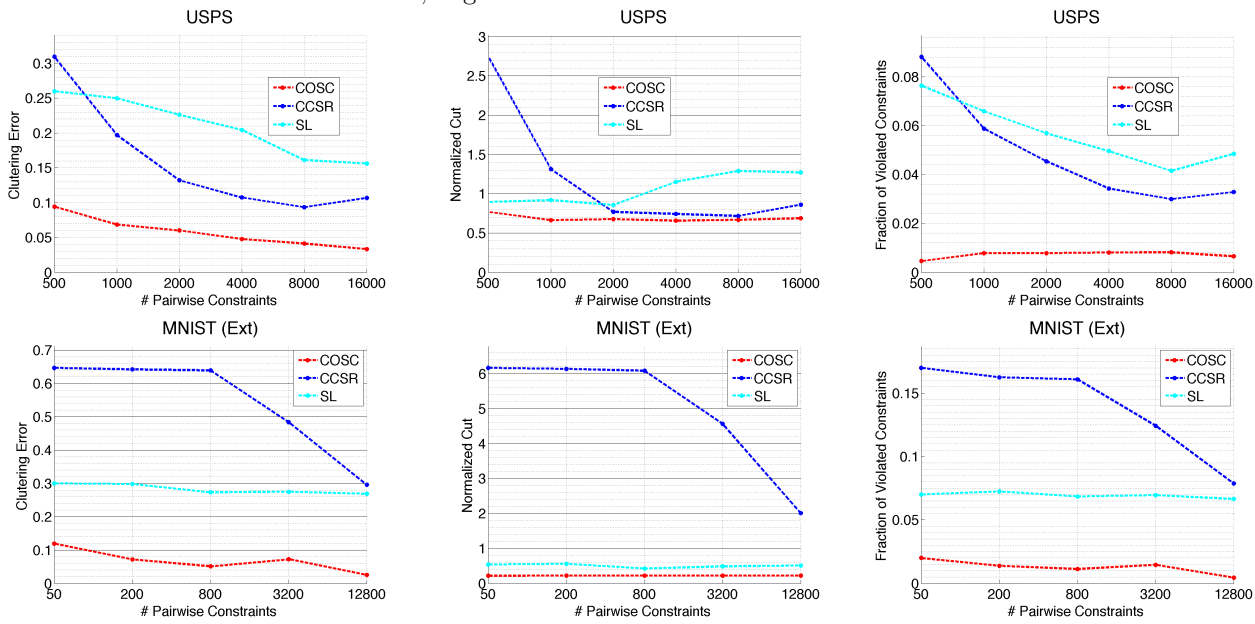


Table 3: Results for **multi-partitioning** - Left: clustering error versus number of constraints, Middle: normalized cut versus number of constraints, Right: fraction of violated constraints versus number of constraints.



References

Basu, S., Davidson, I., & Wagstaff, K. (2008). *Constrained clustering: Advances in algorithms, theory, and applications*. Chapman & Hall.

Beck, A., & Teboulle, M. (2009). Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, *18*, 2419–2434.

Bühler, T., & Hein, M. (2009). Spectral clustering based on the graph p-Laplacian. *Proc. 26th Int. Conf. on Machine Learning (ICML)* (pp. 81–88). Omnipress.

Coleman, T., Saunderson, J., & Wirth, A. (2008). Spectral clustering with inconsistent advice. *Proceedings of the 25th international conference on Machine learning* (pp. 152–159). New York, NY, USA: ACM.

Eriksson, A., Olsson, C., & Kahl, F. (2007). Normalized cuts revisited: A reformulation for segmentation with linear grouping constraints. *IEEE 11th International Conference on Computer Vision (ICCV)* (pp. 1–8).

Guattery, S., & Miller, G. L. (1998). On the quality of spectral separators. *SIAM Journal on Matrix Analysis and Applications*, *19*, 701–719.

Hagen, L. W., & Kahng, A. B. (1991). Fast spectral methods for ratio cut partitioning and clustering. *Proc. Internat. Conf. on Computer Aided Design* (pp. 10–13).

Hein, M., & Bühler, T. (2010). An inverse power method for nonlinear eigenproblems with applications in 1-spectral clustering and sparse PCA. *Advances in Neural Information Processing Systems 23 (NIPS)* (pp. 847–855).

Hein, M., & Setzer, S. (2011). Beyond spectral clustering - tight relaxations of balanced graph cuts. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira and K. Weinberger (Eds.), *Advances in neural information processing systems 24*, 2366–2374.

Hiriart-Urruty, J.-B., & Lemaréchal, C. (2001). *Fundamentals of convex analysis*. Berlin: Springer.

Kamvar, S., Klein, D., & Manning, C. (2003). Spectral learning. *Proc. of the 18th International Joint Conference On Artificial Intelligence (IJCAI)* (pp. 561–566). Morgan Kaufmann.

Li, Z., Liu, J., & Tang, X. (2009). Constrained clustering via spectral regularization. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 421–428).

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *22*, 888–905.

Szlam, A., & Bresson, X. (2010). Total variation and Cheeger cuts. *Proc. of the 27th International Conference on Machine Learning (ICML)* (pp. 1039–1046). Omnipress.

von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, *17*, 395–416.

Wagstaff, K., Cardie, C., Rogers, S., & Schrödl, S. (2001). Constrained k-means clustering with background knowledge. *Proc. of the 18th International Conference on Machine Learning (ICML)* (pp. 577–584).

Wang, X., & Davidson, I. (2010). Flexible Constrained Spectral Clustering. *Proc. of the 16th ACM SIGKDD International conference on Knowledge Discovery and Data mining (KDD)* (pp. 563–572). ACM Press.

Xu, L., Li, W., & Schuurmans, D. (2009). Fast normalized cut with linear constraints. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 421–428).

Yu, S. X., & Shi, J. (2004). Segmentation given partial grouping constraints. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *26*, 173–183.