Quadratic Decomposable Submodular Function Minimization: Theory and Practice

Pan Li   PANLI@PURDUE.EDU
Department of Computer Science
Purdue University
West Lafayette, IN 47907, USA

Niao He   NIAOHE@ILLINOIS.EDU
Department of Industrial and Enterprise Systems Engineering
University of Illinois at Urbana-Champaign
Champaign, IL 61820, USA

Olgica Milenkovic   MILENKOV@ILLINOIS.EDU
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Champaign, IL 61820, USA

Abstract

We introduce a new convex optimization problem, termed *quadratic decomposable submodular function minimization* (QDSFM), which allows to model a number of learning tasks on graphs and hypergraphs. The problem exhibits close ties to decomposable submodular function minimization (DSFM) yet is much more challenging to solve. We approach the problem via a new dual strategy and formulate an objective that can be optimized through a number of double-loop algorithms. The outer-loop uses either random coordinate descent (RCD) or alternative projection (AP) methods, for both of which we prove linear convergence rates. The inner-loop computes projections onto cones generated by base polytopes of the submodular functions via the modified min-norm-point or Frank-Wolfe algorithms. We also describe two new applications of QDSFM: hypergraph-adapted PageRank and semi-supervised learning. The proposed hypergraph-based PageRank algorithm can be used for local hypergraph partitioning and comes with provable performance guarantees. For hypergraph-adapted semi-supervised learning, we provide numerical experiments demonstrating the efficiency of our QDSFM solvers and their significant improvements on prediction accuracy when compared to state-of-the-art methods.

Keywords: Submodular functions, Lovász extensions, Random coordinate descent, Frank-Wolfe method, PageRank.

1. Introduction

Given $[N] = \{1, 2, ..., N\}$, a submodular function $F : 2^{[N]} \to \mathbb{R}$ is a set function that for any $S_1, S_2 \subseteq [N]$ satisfies $F(S_1) + F(S_2) \geq F(S_1 \cup S_2) + F(S_1 \cap S_2)$ (Edmonds, 2003). Submodular functions are ubiquitous in machine learning as they capture rich combinatorial properties of set functions and provide useful regularizers for supervised and unsupervised

learning objectives (Bach, 2013). Submodular functions are endowed with convex Lovász extensions (Lovász, 1983), which provide the means to establish connections between combinatorial and continuous optimization problems (Fujishige, 2005).

Due to their versatility, submodular functions and their Lovász extensions are frequently used in applications such as learning on directed/undirected graphs and hypergraphs (Zhu et al., 2003a; Zhou et al., 2004; Hein et al., 2013), image denoising via total variation regularization (Osher et al., 2005; Chambolle and Darbon, 2009), and maximum a posteriori probability (MAP) inference in high-order Markov random fields (Kumar and Hebert, 2003). In many optimization settings involving submodular functions, one encounters the convex program

$$\min_x \sum_{i \in [N]} (x_i - a_i)^2 + \sum_{r \in [R]} [f_r(x)]^p, \tag{1}$$

where $a \in \mathbb{R}^N$, $p \in \{1, 2\}$, and for all r in some index set $[R]$, f_r is the Lovász extension of a submodular function F_r that describes some combinatorial structure over the set $[N]$. For example, in image denoising, the parameter a_i may correspond to the observed value of pixel i, while the functions $[f_r(x)]^p$ may be used to impose smoothness constraints on the pixel neighborhoods. One of the main difficulties in solving the above optimization problem comes from the nondifferentiability of the second term: a direct application of subgradient methods leads to convergence rates as slow as $1/\sqrt{k}$, where k denotes the number of iterations (Shor et al., 1985).

In recent years, the above optimization problem for $p = 1$, referred to as *decomposable submodular function minimization (DSFM) (Stobbe and Krause, 2010)*, has received significant attention. The motivation for studying this particular setup is two-fold: first, solving the convex optimization problem directly recovers the combinatorial solution to the submodular min-cut problem (Fujishige, 2005) which takes the form $\min_{S \subseteq [N]} F(S)$, where

$$F(S) = \sum_{r \in [R]} F_r(S) - 2 \sum_{i \in S} a_i.$$

Second, minimizing a submodular function decomposed into a sum of simpler components F_r, $r \in [R]$, is significantly easier than minimizing an unrestricted submodular function F over a large set $[N]$. There are several milestone results for solving the DSFM problem: Jegelka et al. (2013) first tackled the problem by considering its dual and proposed a solver based on Douglas-Rachford splitting. Nishihara et al. (2014) established linear convergence rates of alternating projection methods for solving the dual problem. Ene and Nguyen (2015); Ene et al. (2017) presented linear convergence rates of coordinate descent methods and subsequently tightened known results via the use of submodular flows. The work in Li and Milenkovic (2018a) improved previous methods by leveraging incidence relations between the arguments of submodular functions and their constituent parts.

We focus on the case $p = 2$ and refer to the underlying optimization problem as *quadratic DSFM* (QDSFM). QDSFM appears naturally in a wide spectrum of applications, including learning on graphs and hypergraphs, and in particular, semi-supervised learning and PageRank analysis, to be described in more detail later. Moreover, it has been demonstrated both theoretically (Johnson and Zhang, 2007) and empirically (Zhou et al., 2004; Hein et al.,

2013) that using quadratic regularizers in (1) improves the predictive performance for semi-supervised learning when compared to the case $p = 1$. Despite the importance of QDSFM, it has not received the same level of attention as DSFM, neither from the theoretical nor algorithmic perspective. To our best knowledge, only a few reported works (Hein et al., 2013; Zhang et al., 2017) provided solutions for *specific instances* of QDSFM with sublinear convergence guarantees. For the general QDSFM problem, no analytical results are currently known.

Our work takes a substantial step towards solving the QDSFM problem in its most general form by developing a family of algorithms with *linear convergence rates* and *small iteration cost*, including the randomized coordinate descent (RCD) and alternative projection (AP) algorithms. Our contributions are as follows.

1. First, we derive a new dual formulation for the QDSFM problem since an analogue of the dual transformation for the DSFM problem is not applicable. Interestingly, the dual QDSFM problem requires one to find the best approximation of a hyperplane via a product of cones as opposed to a product of polytopes, encountered in the dual DSFM problem.

2. Second, we develop linearly convergent RCD and AP algorithms for solving the dual QDSFM problem. Because of the special underlying conic structure, a new analytic approach is needed to prove the weak strong-convexity of the dual QDSFM problem, which essentially guarantees linear convergence.

3. Third, we develop generalized Frank-Wolfe (FW) and min-norm-point methods for efficiently computing the conic projection required in each step of RCD and AP and provide a $1/k$-rate convergence analysis. These FW-type algorithms and their corresponding analyses do not rely on the submodularity assumption and apply to general conic projections, and are therefore of independent interest.

4. Fourth, we introduce a novel application of the QDSFM problem, related to the PageRank (PR) process for hypergraphs. The main observation is that PR state vectors can be efficiently computed by solving QDSFM problems. We also show that many important properties of the PR process for graphs carry over to the hypergraph setting, including mixing and local hypergraph partitioning results.

5. Finally, we evaluate our methods on semi-supervised learning over hypergraphs using synthetic and real datasets, and demonstrate superior performance both in convergence rate and prediction accuracy compared to existing general-purpose methods.

2. Problem Formulations

We first introduce some terminology and notation useful for our subsequent exposition.

Lovász extension and base polytope. For a submodular function F defined over the ground set $[N]$, the *Lovász extension* is a convex function $f : \mathbb{R}^N \to \mathbb{R}$, defined for all

$x \in \mathbb{R}^N$ according to

$$f(x) = \sum_{k=1}^{N-1} F(\{i_1, ..., i_k\})(x_{i_k} - x_{i_{k+1}}) + F([N])x_{i_N}, \tag{2}$$

where $x_{i_1} \geq x_{i_2} \geq \cdots \geq x_{i_N}$, and $i_1, ..., i_N$ are distinct indices in $[N]$. For a vector $x \in \mathbb{R}^N$ and a set $S \subseteq [N]$, define $x(S) = \sum_{i \in S} x_i$, where x_i stands for the value of the i-th dimension of x. The *base polytope* of F, denoted by B, is defined as

$$B = \{y \in \mathbb{R}^N | y(S) \leq F(S), \ \forall S \subset [N], \ y([N]) = F([N])\}. \tag{3}$$

Using the base polytope, the Lovász extension can also be written as $f(x) = \max_{y \in B}\langle y, x\rangle$. We say that an element $i \in [N]$ is *incident* to F if there exists a $S \subset [N]$ such that $F(S) \neq F(S \cup \{i\})$.

Notations. Given a positive diagonal matrix $W \in \mathbb{R}^{N \times N}$ and a vector $x \in \mathbb{R}^N$, we define the W-norm according to $\|x\|_W = \sqrt{\sum_{i=1}^N W_{ii}\, x_i^2}$, and simply use $\|\cdot\|$ when $W = I$, the identity matrix. For an index set $[R]$, we denote the R-product of N-dimensional Euclidean spaces by $\otimes_{r \in [R]}\mathbb{R}^N$. A vector $y \in \otimes_{r \in [R]}\mathbb{R}^N$ is written as $(y_1, y_2, ..., y_R)$, where $y_r \in \mathbb{R}^N$ for all $r \in [R]$. The W-norm induced on $\otimes_{r \in [R]}\mathbb{R}^N$ equals $\|y\|_{I(W)} = \sqrt{\sum_{r=1}^R \|y_r\|_W^2}$. We reserve the symbol ρ for $\max_{y_r \in B_r, \forall r} \sqrt{\sum_{r \in [R]} \|y_r\|_1^2}$. Furthermore, we use $(x)_+$ to denote the function $\max\{x, 0\}$.

2.1 The QDSFM Problem

Consider a collection of submodular functions $\{F_r\}_{r \in [R]}$ defined over the ground set $[N]$, and denote their Lovász extensions and base polytopes by $\{f_r\}_{r \in [R]}$ and $\{B_r\}_{r \in [R]}$, respectively. Let $S_r \subseteq [N]$ denote the set of variables incident to F_r and assume that the functions F_r are normalized and nonnegative, i.e., that $F_r(\emptyset) = 0$ and $F_r \geq 0$. These two mild constraints are satisfied by almost all submodular functions that arise in practice. Note that we do not impose monotonicity constraints which are frequently assumed and used in submodular function optimization; nevertheless, our setting naturally applies to the special case of monotonously increasing submodular functions.

The QDSFM problem is formally stated as follows:

$$\text{QDSFM:} \qquad \min_{x \in \mathbb{R}^N} \|x - a\|_W^2 + \sum_{r \in [R]} [f_r(x)]^2, \tag{4}$$

where $a \in \mathbb{R}^N$ is a given vector and $W \in \mathbb{R}^{N \times N}$ is a positive diagonal matrix. The QDSFM problem is convex because the Lovász extensions f_r are convex, and nonnegative. In fact, it is also strongly convex, implying that there exists a unique optimal solution, denoted by x^*. The quadratic form $[f_r(x)]^2$ may be generalized to $q(f_r(x))$, where the mapping $q(\cdot) : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ satisfies the monotonicity, strong convexity and smoothness conditions. In this case, subsequent results such as those pertaining to the transformation (5) may be obtained by using the convex conjugate of $q(\cdot)$, leading to similar conclusions as those

described in Section 3 for the quadratic form. Our focus is on quadratic forms as they correspond to important applications described in Sections 5 and 6.

As a remark, in contrast to the DSFM problem, we are not aware of any discrete optimization problems that directly correspond to the QDSFM problem. However, as we will show in Section 5, QDSFM can be used to compute the PageRank vectors for hypergraphs which itself is a relevant discrete optimization question. Moreover, as will be demonstrated in subsequent sections, the obtained PageRank vectors are helpful for approximating the hypergraph conductance, another important discrete optimization objective.

2.2 Dual Formulation

Despite being convex, the objective is in general nondifferentiable. This implies that only sublinear convergence can be obtained when directly applying the subgradient method. To address this issue, we revert to the dual formulation of the QDSFM problem. A natural idea is to mimic the approach used for DSFM by exploiting the characterization of the Lovász extension, $f_r(x) = \max_{y_r \in B_r} \langle y_r, x \rangle$, $\forall r$. However, this leads to a semidefinite programing problem for the dual variables $\{y_r\}_{r \in [R]}$, which is too expensive to solve for large problems. Instead, we establish a new dual formulation that overcomes this obstacle.

The dual formulation hinges upon the following simple yet key observation:

$$[f_r(x)]^2 = \max_{\phi_r \geq 0} \phi_r f_r(x) - \frac{\phi_r^2}{4} = \max_{\phi_r \geq 0} \max_{y_r \in \phi_r B_r} \langle y_r, x \rangle - \frac{\phi_r^2}{4}. \tag{5}$$

Let $y = (y_1, y_2, ..., y_R)$ and $\phi = (\phi_1, \phi_2, ..., \phi_R)$. For each r, we define a convex cone

$$C_r = \{(y_r, \phi_r) | \phi_r \geq 0, y_r \in \phi_r B_r\}$$

which represents the feasible set of the variables (y_r, ϕ_r). Furthermore, we denote the product of cones by

$$\mathcal{C} = \bigotimes_{r \in [R]} C_r := \{(y, \phi) : \phi_r \geq 0, y_r \in \phi_r B_r, \forall r \in [R]\}.$$

Invoking equation (5), we arrive at the following two dual formulations for the original QDSFM problem in Lemma 1.

Lemma 1 *The following optimization problem is dual to (4):*

$$\min_{y,\phi} g(y, \phi) := \| \sum_{r \in [R]} y_r - 2Wa \|_{W^{-1}}^2 + \sum_{r \in [R]} \phi_r^2, \ s.t. \ (y, \phi) \in \mathcal{C}. \tag{6}$$

By introducing $\Lambda = (\lambda_r) \in \bigotimes_{r \in [R]} \mathbb{R}^N$, the previous optimization problem takes the form

$$\min_{y,\phi,\Lambda} \sum_{r \in [R]} \left[\|y_r - \frac{\lambda_r}{\sqrt{R}}\|_{W^{-1}}^2 + \phi_r^2 \right], \ s.t. \ (y, \phi) \in \mathcal{C}, \sum_{r \in [R]} \lambda_r = 2Wa. \tag{7}$$

The primal variables in both cases may be computed as $x = a - \frac{1}{2} W^{-1} \sum_{r \in [R]} y_r$.

The proof of Lemma 1 is presented in Section 7.1.1. The dual formulations for the DSFM problem were described in Lemma 2 of (Jegelka et al., 2013). Similarly to the DSFM problem, the dual formulations (6) and (7) are simple, nearly separable in the dual variables, and may be solved by algorithms such as those used for solving dual DSFM, including the Douglas-Rachford splitting method (DR) (Jegelka et al., 2013), the alternative projection method (AP) (Nishihara et al., 2014), and the random coordinate descent method (RCD) (Ene and Nguyen, 2015).

However, there are also some notable differences. Our dual formulations for the QDSFM problem are defined on a product of cones constructed from the base polytopes of the submodular functions. The optimization problem (7) essentially asks for the best approximation of an affine space in terms of the product of cones, whereas in the DSFM problem one seeks an approximation in terms of a product of polytopes. In the next section, we propose to solve the dual problem (6) using the random coordinate descent method (RCD), and to solve (7) using the alternative projection method (AP), both tailored to the conic structure. The conic structures make the convergence analysis and the computations of projections for these algorithms challenging.

Before proceeding with the description of our algorithms, we first provide a relevant geometric property of the product of cones $\bigotimes_{r \in [R]} C_r$, as stated in the following lemma. This property is essential in establishing the linear convergence of RCD and AP algorithms.

Lemma 2 *Consider a feasible solution of the problem* $(y, \phi) \in \bigotimes_{r \in [R]} C_r$ *and a nonnegative vector* $\phi' = (\phi'_r) \in \bigotimes_{r \in [R]} \mathbb{R}_{\geq 0}$. *Let* s *be an arbitrary point in the base polytope of* $\sum_{r \in [R]} \phi'_r F_r$, *and let* $W^{(1)}, W^{(2)}$ *be two positive diagonal matrices. Then, there exists a* $y' \in \bigotimes_{r \in [R]} \phi'_r B_r$ *satisfying* $\sum_{r \in [R]} y'_r = s$ *and*

$$\|y - y'\|^2_{I(W^{(1)})} + \|\phi - \phi'\|^2 \leq \mu(W^{(1)}, W^{(2)}) \left[\| \sum_{r \in [R]} y_r - s\|^2_{W^{(2)}} + \|\phi - \phi'\|^2 \right],$$

where

$$\mu(W^{(1)}, W^{(2)}) = \max \left\{ \sum_{i \in [N]} W^{(1)}_{ii} \sum_{j \in [N]} 1/W^{(2)}_{jj}, \frac{9}{4}\rho^2 \sum_{i \in [N]} W^{(1)}_{ii} + 1 \right\}, \quad (8)$$

and $\rho = \max_{y_r \in B_r, \forall r \in [R]} \sqrt{\sum_{r \in [R]} \|y_r\|^2_1}$.

The proof of Lemma 2 is relegated to Section 7.1.2.

3. Linearly Convergent Algorithms for Solving the QDSFM Problem

Next, we introduce and analyze the random coordinate descent method (RCD) for solving the dual problem (6), and the alternative projection method (AP) for solving (7). Both methods exploit the separable structure of the feasible set. It is worth mentioning that our results may be easily extended to the Douglas-Rachford splitting method as well as accelerated and parallel variants of the RCD method used to solve DSFM problems (Jegelka et al., 2013; Ene and Nguyen, 2015; Li and Milenkovic, 2018a).

Algorithm 1: The RCD method for Solving (6)

0: For all r, initialize $y_r^{(0)} \leftarrow 0$, $\phi_r^{(0)}$ and $k \leftarrow 0$

1: In iteration k:

2:     Uniformly at random pick an $r \in [R]$.

3:     $(y_r^{(k+1)}, \phi_r^{(k+1)}) \leftarrow \Pi_{C_r}(2Wa - \sum_{r' \neq r} y_{r'}^{(k)}; W^{-1})$

4:     Set $y_{r'}^{(k+1)} \leftarrow y_{r'}^{(k)}$ for $r' \neq r$

We define the projection Π onto a convex cone C_r as follows: for a point b in $\mathbb{R}^N$ and a positive diagonal matrix W' in $\mathbb{R}^{N \times N}$, we set

$$\Pi_{C_r}(b; W') = \operatorname*{argmin}_{(y_r, \phi_r) \in C_r} \|y_r - b\|_{W'}^2 + \phi_r^2.$$

Throughout the remainder of this section, we treat the projections as provided by an oracle. Later in Section 4, we provide some efficient methods for computing the projections.

3.1 The Random Coordinate Descent (RCD) Algorithm

Consider the dual formulation (6). For each coordinate r, optimizing over the dual variables (y_r, ϕ_r) is equivalent to computing a projection onto the cone C_r. This gives rise to the RCD method of Algorithm 1.

The objective $g(y, \phi)$ described in (6) is not strongly convex in general. However, with some additional work, Lemma 2 can be used to establish the weak strong convexity of $g(y, \phi)$; this essentially guarantees a linear convergence rate of the RCD algorithm. To proceed, we need some additional notation. Denote the set of solutions of (6) by

$$\Xi = \{(y, \phi) | \sum_{r \in [R]} y_r = 2W(a - x^*), \phi_r = \inf_{y_r \in \theta B_r} \theta, \forall r\}. \tag{9}$$

Note that the optimal ϕ_r, as defined in (9), is closely related to gauge functions of polytopes (Bach, 2013). This representation is a consequence of the relationship between the optimal primal and dual solutions stated in Lemma 1. For convenience, we denote the optimal value of the objective function of interest over $(y, \phi) \in \Xi$ by $g^* = g(y, \phi)$, and also define a distance function

$$d((y, \phi), \Xi) = \sqrt{\min_{(y', \phi') \in \Xi} \|y - y'\|_{I(W^{-1})}^2 + \|\phi - \phi'\|^2}.$$

Lemma 3 (Weak Strong Convexity) *Suppose that* $(y, \phi) \in \bigotimes_{r \in [R]} C_r$ *and that* $(y^*, \phi^*) \in \Xi$ *minimizes* $\|y - y'\|_{I(W^{-1})}^2 + \|\phi - \phi'\|^2$ *over* $(y', \phi') \in \Xi$, *i.e.,* (y^*, ϕ^*) *is the projection of* (y, ϕ) *onto* Ξ. *Then,*

$$\| \sum_{r \in [R]} (y_r - y_r^*)\|_{W^{-1}}^2 + \|\phi - \phi^*\|^2 \geq \frac{d^2((y, \phi), \Xi)}{\mu(W^{-1}, W^{-1})}. \tag{10}$$

Lemma 3 can be proved by applying Lemma 2 with $\phi' = \phi^*, s = 2W(a-x^*), W^{(1)}, W^{(2)} = W^{-1}$ and the definition of y^*. Note that the claim in (10) is significantly weaker than the usual strong convexity condition. We show that such a condition is sufficient to ensure linear convergence of the RCD algorithm and provides the results in the following theorem.

Theorem 4 (Linear Convergence of RCD) *Running k iterations of Algorithm 1 produces a pair $(y^{(k)}, \phi^{(k)})$ that satisfies*

$$\mathbb{E}\left[g(y^{(k)}, \phi^{(k)}) - g^* + d^2((y^{(k)}, \phi^{(k)}), \Xi)\right]$$

$$\leq \left[1 - \frac{2}{R[1 + \mu(W^{-1}, W^{-1})]}\right]^k \left[g(y^{(0)}, \phi^{(0)}) - g^* + d^2((y^{(0)}, r^{(0)}), \Xi)\right].$$

Theorem 4 asserts that at most $O(R\mu(W^{-1}, W^{-1}) \log \frac{1}{\epsilon})$ iterations are required to obtain an ϵ-optimal solution in expectation for the QDSFM problem. The proof of Lemma 3 and Theorem 4 are postponed to Section 7.2.

3.2 The Alternative Projection (AP) Algorithm

The AP method can be used to solve the dual problem (7), which is of the form of a *best-approximation problem*, by alternatively performing projections between the product of cones and a hyperplane. Furthermore, for some incidence relations, S_r may be a proper subset of $[N]$, which consequently requires the ith component of y_r, i.e., $y_{r,i}$ to be zero if $i \notin S_r$. Enforcing $\lambda_{r,i} = 0$ for $i \notin S_r$ allows the AP method to avoid redundant computations and achieve better convergence rates. This phenomenon has also been observed for the DSFM problem in (Djolonga and Krause, 2015; Li and Milenkovic, 2018a). To avoid redundant computations, we use the AP approach to solve the following dual problem:

$$\min_{y,\phi,\Lambda} \quad \sum_{r\in[R]} \left[\|y_r - \lambda_r\|^2_{\Psi W^{-1}} + \phi_r^2\right], \tag{11}$$

$$\text{s.t. } (y, \phi) \in \mathcal{C}, \sum_{r\in[R]} \lambda_r = 2Wa, \text{ and } \lambda_{r,i} = 0 \text{ for all } i \notin S_r.$$

Here, $\Psi \in \mathbb{R}^{N\times N}$ is a positive diagonal matrix in which $\Psi_{ii} = |\{r \in [R] | i \in S_r\}|$ equals the number of submodular functions that i is incident to.

Lemma 5 *Problem (11) is equivalent to problem (6).*

The proof of Lemma 5 is presented in Section 7.3.1. The AP method for solving (11) is listed in Algorithm 2. Observe that Step 5 is a projection onto cones defined based on the positive diagonal matrix ΨW^{-1} which differs from the one used in the RCD method. Note that compared to the RCD algorithm, AP requires one to compute projections onto *all* cones C_r in each iteration. Thus, in most cases, AP requires larger computation cost than the RCD algorithm. On the other hand, AP naturally lends itself to parallelization since the projections can be decoupled and implemented very efficiently.

Before moving to the convergence analysis, we first prove the uniqueness of optimal ϕ in the following lemma.

8

Algorithm 2: The AP Method for Solving (11)

0: For all r, initialize $y_r^{(0)} \leftarrow 0, \phi_r^{(0)} \leftarrow 0$, and $k \leftarrow 0$

1: In iteration k:

2: $\quad \alpha^{(k+1)} \leftarrow 2W^{-1} \sum_r y_r^{(k)} - 4a.$

3: $\quad$ For all $r \in [R]$:

4: $\qquad \lambda_{r,i}^{(k+1)} \leftarrow y_{r,i}^{(k)} - \frac{1}{2}(\Psi^{-1}W\alpha^{(k+1)})_i$ for $i \in S_r$

5: $\qquad (y_r^{(k+1)}, \phi_r^{(k+1)}) \leftarrow \Pi_{C_r}(\lambda_r^{(k+1)}; \Psi W^{-1})$

Lemma 6 *The optimal value of ϕ to problem (11) is unique.*

Proof We know (11) is equivalent to (6). Suppose there are two optimal solutions $(\bar{y}, \bar{\phi})$ and $(\tilde{y}, \tilde{\phi})$. As the optimal x^* is unique, $\sum_{r \in [R]} \bar{y}_r = \sum_{r \in [R]} \tilde{y}_r = 2W(a - x^*) = \sum_{r \in [R]} \frac{\bar{y}_r + \tilde{y}_r}{2}$. If $\bar{\phi} \neq \tilde{\phi}$, we have $\sum_{r \in [R]} \left(\frac{\bar{\phi}_r + \tilde{\phi}_r}{2} \right)^2 < \sum_{r \in [R]} \bar{\phi}_r^2 = \sum_{r \in [R]} \tilde{\phi}_r^2$, which makes $g(\frac{\bar{y} + \tilde{y}}{2}, \frac{\bar{\phi} + \tilde{\phi}}{2})$ smaller than $g(\bar{y}, \bar{\phi}) = g(\tilde{y}, \tilde{\phi})$ and thus causes contradiction. ∎

To determine the convergence rate of the AP method, we adapt the result of (Nishihara et al., 2014) on the convergence rate of APs between two convex bodies. In our setting, the two convex bodies of interest are the the cone $\mathcal{C}$ and the hyperplane

$$\mathcal{Z} = \{(y, \phi) | \sum_{r \in [R]} y_r = 2W(a - x^*), \phi_r = \phi_r^*, y_{r,i} = 0, \forall i \notin S_r\},$$

where $\phi^* = (\phi_r^*)_{r \in [R]}$ is the unique optimal solution of (11).

Lemma 7 (Nishihara et al. (2014)) *Let Ξ be as defined in (9). In addition, define the distance function*

$$d_{\Psi W^{-1}}((y, \phi), \Xi) = \sqrt{\min_{(y', \phi') \in \Xi} \|y - y'\|_{I(\Psi W^{-1})}^2 + \|\phi - \phi'\|^2}.$$

In the k-th iteration of Algorithm 5, the pair $(y^{(k)}, \phi^{(k)})$ satisfies

$$d_{\Psi W^{-1}}((y^{(k)}, \phi^{(k)}), \Xi) \leq 2d_{\Psi W^{-1}}((y^{(0)}, \phi^{(0)}), \Xi)(1 - \frac{1}{\kappa_*^2})^k,$$

where

$$\kappa_* = \sup_{(y, \phi) \in \mathcal{C} \cup \mathcal{Z}/\Xi} \frac{d_{\Psi W^{-1}}((y, \phi), \Xi)}{\max\{d_{\Psi W^{-1}}((y, \phi), \mathcal{C}), d_{\Psi W^{-1}}((y, \phi), \mathcal{Z})\}}.$$

The next Lemma establishes a finite upper bound on κ_*. This guaratees linear convergence rates for the AP algorithm.

Lemma 8 *One has $\kappa_*^2 \leq 1 + \mu(\Psi W^{-1}, W^{-1})$.*

The proof of Lemma 8 may be found in Section 7.3.2.

Lemma 8 implies that running the AP algorithm with $O(\mu(\Psi W^{-1}, W^{-1}) \log \frac{1}{\epsilon})$ iterations guarantee that $d_{\Psi W^{-1}}((y^{(k)}, \phi^{(k)}), \Xi) \leq \epsilon$. Moreover, as $\Psi_{ii} \leq R$, the iteration complexity

of the AP method is smaller than that of the RCD algorithm. However, each iteration of the AP solver requires performing projections on all cones C_r, $r \in [R]$, while each iteration of the RCD method requires computing only one projection onto a single cone. Other methods such as the DR and primal-dual hybrid gradient descent (PDHG) (Chambolle and Pock, 2011) used in semi-supervised learning on hypergraphs, which is a specific QDSFM problem (Hein et al., 2013), also require computing a total of R projections during each iteration. Thus, from the perspective of iteration cost, RCD is significantly more efficient, especially when R is large and computing $\Pi(\cdot)$ is costly. This phenomenon is also observed in practice, as illustrated by the experiments in Section 6.1.

The following corollary summarizes the previous discussion and will be used in our subsequent derivations in Section 5.1.

Corollary 9 *Suppose that $W = \beta D$, where β is a hyper-parameter, and D is a diagonal matrix such that*

$$D_{ii} = \sum_{r \in [R]: i \in S_r} \max_{S \subseteq V} [F_r(S)]^2.$$

Recall that $\Psi_{ii} = |\{r \in [R] | i \in S_r\}|$. Then, Algorithm 1 (RCD) requires an expected number of

$$O\left(N^2 R \max\{1, 9\beta^{-1}\} \max_{i,j \in [N]} \frac{D_{ii}}{D_{jj}} \log \frac{1}{\epsilon} \right)$$

iterations to return a solution (y, ϕ) that satisfies $d((y, \phi), \Xi) \leq \epsilon$. Algorithm 2 (AP) requires a number of

$$O\left(N^2 R \max\{1, 9\beta^{-1}\} \max_{i,j \in [N]} \frac{\Psi_{jj}}{R} \frac{D_{ii}}{D_{jj}} \log \frac{1}{\epsilon} \right)$$

iterations to return a solution (y, ϕ) that satisfies $d_{\Psi W^{-1}}((y, \phi), \Xi) \leq \epsilon$.

The proof of Corollary 9 is provided in Section 7.4. Note that the term $N^2 R$ also appears in the expression for the complexity of the RCD and the AP methods for solving the DSFM problem (Ene et al., 2017). The term $\max\{1, 9\beta^{-1}\}$ implies that whenever β is small, the convergence rate is slow. In Section 5.1, we provide a case-specific analysis showing that β is related to the mixing time of the underlying Markov process in the PageRank application. Depending on the application domain, other interpretations of β could exist, but will not be discussed here to avoid clutter. It is also worth pointing out that the variable D in Corollary 9 is closely related to the notion of degree in hypergraphs (Li and Milenkovic, 2018b; Yoshida, 2019).

4. Computing the Conic Projections

In this section, we describe efficient routines for computing a projection onto the conic set $\Pi_{C_r}(\cdot)$. As the procedure works for all values of $r \in [R]$, we drop the subscript r for simplicity of notation. Let $C = \{(y, \phi) | y \in \phi B, \phi \geq 0\}$, where B denotes the base polytope of the submodular function F. Recall that the conic projection onto C is defined as

$$\Pi_C(a; \tilde{W}) = \arg \min_{(y, \phi)} h(y, \phi) \triangleq \|y - a\|_{\tilde{W}}^2 + \phi^2 \quad \text{s.t.} \quad (y, \phi) \in C. \tag{12}$$

Algorithm 3: The Conic MNP Method for Solving (12)

Input: $\tilde{W}$, a, B and a small positive constant δ. **Maintain** $\phi^{(k)} = \sum_{q_i \in S^{(k)}} \lambda_i^{(k)}$

Choose an arbitrary $q_1 \in B$. Set $S^{(0)} \leftarrow \{q_1\}$, $\lambda_1^{(0)} \leftarrow \frac{\langle a, q_1 \rangle_{\tilde{W}}}{1 + \|q_1\|_{\tilde{W}}^2}$, $y^{(0)} \leftarrow \lambda_1^{(0)} q_1$, $k \leftarrow 0$

1. Iteratively execute (**MAJOR LOOP**):
2. $q^{(k)} \leftarrow \arg\min_{q \in B} \langle \nabla_y h(y^{(k)}, \phi^{(k)}), q \rangle_{\tilde{W}}$
3.   **If** $\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} \geq -\delta$, **break**;
4.   **Else** $S^{(k)} \leftarrow S^{(k)} \cup \{q^{(k)}\}$.
5.   Iteratively execute (**MINOR LOOP**):
6. $\alpha \leftarrow \arg\min_\alpha \|\sum_{q_i^{(k)} \in S^{(k)}} \alpha_i q_i^{(k)} - a\|_{\tilde{W}}^2 + (\sum_{q_i^{(k)} \in S} \alpha_i)^2$,
7. $z^{(k)} \leftarrow \sum_{q_i^{(k)} \in S} \alpha_i q_i^{(k)}$
8.   **If** $\alpha_i \geq 0$ for all i, **break**;
9.   **Else** $\theta = \min_{i:\alpha_i < 0} \lambda_i^{(k)} / (\lambda_i^{(k)} - \alpha_i)$, $\lambda_i^{(k+1)} \leftarrow \theta \alpha_i + (1 - \theta) \lambda_i^{(k)}$,
10. $y^{(k+1)} \leftarrow \theta z^{(k)} + (1 - \theta) y^{(k)}$, $S^{(k+1)} \leftarrow \{i : \lambda^{(k+1)} > 0\}$, $k \leftarrow k + 1$
11. $y^{(k+1)} \leftarrow z^{(k)}$, $\lambda^{(k+1)} \leftarrow \alpha$, $S^{(k+1)} \leftarrow \{i : \lambda^{(k+1)} > 0\}$, $k \leftarrow k + 1$

Let h^* and (y^*, ϕ^*) be the optimal value of the objective function and the optimal solution, respectively. When performing projections, one only needs to consider the variables incident to F, and set all other variables to zero. For simplicity, we assume that all variables in $[N]$ are incident to F. The results can easily extend to the case that the incidences are of a general form.

Unlike the QDSFM problem, solving the DSFM involves the computation of projections onto the base polytopes of submodular functions. Two algorithms, the Frank-Wolfe (FW) method (Frank and Wolfe, 1956) and the Fujishige-Wolfe minimum norm algorithm (MNP) (Fujishige and Isotani, 2011), are used for this purpose. Both methods assume inexpensive linear minimization oracles on polytopes and guarantee a $1/k$-convergence rate. The MNP algorithm is more sophisticated yet empirically more efficient. Nonetheless, neither of these methods can be applied directly to conic projections. To this end, we modify these two methods by adjusting them to the conic structure in (12) and show that they still achieve a $1/k$-convergence rate. We refer to the procedures as *the conic FW method* and *the conic MNP method*, respectively.

4.1 The Conic MNP Algorithm

Detailed steps of the conic MNP method are presented in Algorithm 3. The conic MNP algorithm keeps track of an *active set* $S = \{q_1, q_2, ...\}$ and searches for the best solution in its conic hull. Denote the cone of an active set S by $\text{cone}(S) = \{\sum_{q_i \in S} \alpha_i q_i | \alpha_i \geq 0\}$ and its linear set by $\text{lin}(S) = \{\sum_{q_i \in S} \alpha_i q_i | \alpha_i \in \mathbb{R}\}$. Similar to the original MNP algorithm, Algorithm 3 also includes two level-loops: the MAJOR and MINOR loop. In the MAJOR loop, one greedily add a new active point $q^{(k)}$ to the set S obtained from the linear minimization oracle w.r.t. the base polytope (Step 2). By the end of the MAJOR loop, we obtain $y^{(k+1)}$ that minimizes $h(y, \phi)$ over $\text{cone}(S)$ (Step 3-11). The MINOR loop is activated when $\text{lin}(S)$ contains some point z that guarantees a smaller value of the objective function than the optimal point in $\text{cone}(S)$, provided that some active points from S may be removed. It is

---

Algorithm 4: The Conic FW Algorithm for Solving (12)

Input: $\tilde{W}$, a, B and a small positive δ

Initialize $y^{(0)} \leftarrow 0$, $\phi^{(0)} \leftarrow 0$ and $k \leftarrow 0$

1. **Iteratively execute the following steps:**

2. $q^{(k)} \leftarrow \arg\min_{q \in B} \langle \nabla_y h(y^{(k)}, \phi^{(k)}), q \rangle$

3.   **If** $\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} \geq -\delta$, **break**.

4.   **Else**: $(\gamma_1^{(k)}, \gamma_2^{(k)}) \leftarrow \arg\min_{\gamma_1 \geq 0, \gamma_2 \geq 0} h(\gamma_1 y^{(k)} + \gamma_2 q^{(k)}, \gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)})$

5. $y^{(k+1)} \leftarrow \gamma_1^{(k)} y^{(k)} + \gamma_2^{(k)} q^{(k)}$, $\phi^{(k+1)} \leftarrow \gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)}$, $k \leftarrow k + 1$.

---

important to point out that compared to the original MNP method, Steps 2 and 6 as well as the termination Step 3 are specialized for the conic structure.

Theorem 10 below shows that the conic MNP algorithm preserves the $1/k$-convergence rate of the original MNP method.

Theorem 10 *Let B be an arbitrary polytope in $\mathbb{R}^N$ and let $C = \{(y, \phi) | y \in \phi B, \phi \geq 0\}$ be the cone induced by the polytope. For any positive diagonal matrix $\tilde{W}$, define $Q = \max_{q \in B} \|q\|_{\tilde{W}}$. Algorithm 3 produces a sequence of pairs $(y^{(k)}, \phi^{(k)})_{k=1,2,\ldots}$ such that $h(y^{(k)}, \phi^{(k)})$ decreases monotonically. It terminates when $k = O(N\|a\|_{\tilde{W}} \max\{Q^2, 1\}/\delta)$, with*

$$h(y^{(k)}, \phi^{(k)}) \leq h^* + \delta \|a\|_{\tilde{W}}.$$

Note that the result is essentially independent on the submodularity assumption and applies to general cones induced by arbitrary polytopes. Proving Theorem 10 requires a careful modification of the arguments in (Chakrabarty et al., 2014) to handle the conic structure. We provide a detailed proof in Section 7.5.

4.2 The Conic FW Algorithm

We introduce next the conic FW method, summarized in Algorithm 4. Note that Steps 2, 4 are specialized for cones. The main difference between the FW and MNP methods is the size of active set: FW only maintains two active points, while MNP may maintain as many as N points. A similar idea was proposed in (Harchaoui et al., 2015) to deal with the conic constraints, with the following minor difference: our method directly finds the variable $y^{(k+1)}$ from the cone $(\{y^{(k)}, q^{(k)}\})$ (Steps 4, 5) while (Harchaoui et al., 2015) first determines finite upper bounds $(\tilde{\gamma}_1, \tilde{\gamma}_2)$ on the coefficients (γ_1, γ_2) (see the notation in Steps 4, 5) and then searches for the update $y^{(k+1)}$ in $\text{conv}(\{\tilde{\gamma}_1 y^{(k)}, \tilde{\gamma}_2 q^{(k)}, 0\})$. Our method simplifies this procedure by avoiding computations of the upper bounds $(\tilde{\gamma}_1, \tilde{\gamma}_2)$ and removing the constraints $\gamma_1 \leq \tilde{\gamma}_1, \gamma_2 \leq \tilde{\gamma}_2$ in Step 4.

For completeness, we establish a $1/k$-convergence rate for the conic FW algorithm in Theorem 11.

Theorem 11 *Let B be an arbitrary polytope in $\mathbb{R}^N$ and let $C = \{(y, \phi) | y \in \phi B, \phi \geq 0\}$ denote its corresponding cone. For some positive diagonal matrix $\tilde{W}$, define $Q = \max_{q \in B} \|q\|_{\tilde{W}}$. Then, the point $(y^{(k)}, \phi^{(k)})$ generated by Algorithm 4 satisfies*

$$h(y^{(k)}, \phi^{(k)}) \leq h^* + \frac{2\|a\|_{\tilde{W}}^2 Q^2}{k + 2}.$$

The proof of Theorem 11 is deferred to Section 7.6.

4.3 Analysis of Time Complexity and Further Discussion

Let EO stand for the time needed to query the function value of $F(\cdot)$. The linear program in Step 2 of both the MNP and FW algorithms requires $O(N \log N + N \times EO)$ operations if implemented as a greedy method. Step 6 of the MNP method requires solving a quadratic program with no constraints, and the solution may be obtained in closed form using $O(N|S|^2)$ operations. The remaining operations in the MNP method introduce a $O(N)$ complexity term. Hence, the execution of each MAJOR or MINOR loop requires $O(N \log N + N \times EO + N|S|^2)$ operations. In the FW method, the optimization problem in Step 4 is relatively simple, since it reduces to solving a nonnegative quadratic program with only two variables and consequently introduces a complexity term $O(N)$. Therefore, the complexity of the FW algorithm is dominated by Step 2, which requires $O(N \log N + N \times EO)$ operations.

Below we discuss the distinctions of our method from several relevant methods for computing projections on cones. Compared to the linearly convergent away-steps Frank-Wolfe algorithm (Wolfe, 1970) and its generalized versions for the conic case, such as the away-steps/pairwise non-negative matching pursuit (AMP/PWMP) techniques recently proposed in (Locatello et al., 2017), the size of the active set S in the (conic) MNP algorithm is always bounded by N instead of the (increasing) number of iterations. Particularly, when applied to the QDSFM problems, the size of the active set S in the (conic) MNP algorithm can be even smaller in practice as it is bounded by the number of elements that affects the value of the submodular function, which is usually smaller than N (Li and Milenkovic, 2018a). Moreover, in terms of base polytopes of submodular functions, AMP/PWMP may not perform as well as MNP because the steepest ascent over the active set may not be executed as efficiently as the descent over polytopes obtained via the greedy approach (Step 2, which generally appears in FW-based methods).

Another benefit of the (conic) MNP method is that it can compute exact projections for the problem (12) in finitely many steps. Because of the structure of the cones generated from polytopes, the exact projections are expressed as linear combinations of the extreme points of the polytopes. As the number of such combinations is finite, MNP can find the exact projections in finitely many iterations, thereby inheriting a similar property of the MNP method pertaining to polytopes (Fujishige and Isotani, 2011). While this property also holds for the fully corrective non-negative matching pursuit (FCMP) (Locatello et al., 2017), FCMP requires solving an additional nonnegative quadratic program in the inner loop, and is thus much less efficient; MNP does not require an explicit solution of the nonnegative quadratic program as it allows for an efficient update of the set of atoms, $S^{(k)}$, described in Algorithm 3, through proper leveraging of the problem structure (minor loops of Algorithm 3). In contrast, the AMP/PWMP methods in (Locatello et al., 2017) as well as our conic FW method introduced in Section 4.2 may not ensure exact projections. As the proposed algorithms for QDSFM depend highly on precise projections for the problem (12), using MNP as the subroutine for computing projections has the potential to yield better accuracy than other methods discussed above. This is further evidenced in our experiments, to be discussed below.
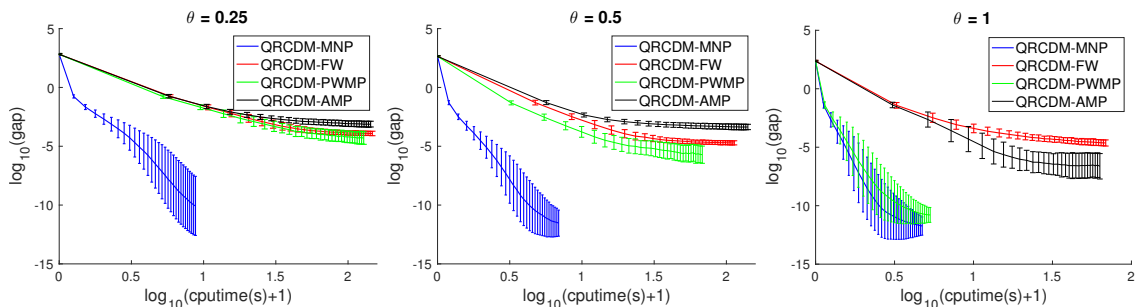
Figure 1: Convergence results ($\log_{10}$(gap): mean $\pm$ std) for the QRCD-MNP, QRCD-FW, QRCD-AMP, QRCD-PWMP algorithms for general submodular functions.

4.4 Numerical Illustration I: Convergence of QRCD-MNP and QRCD-FW

We illustrate the convergence speeds of the RCD algorithm when using the conic MNP and the conic FW methods as subroutines for computing the conic projections, referred to as QRCD-MNP and QRCD-FW, respectively. We also compare them with two other methods, QRCD-AMP and QRCD-PWMP, where the subroutine for computing the conic projection is replaced by AMP and PWMP (Locatello et al., 2017), respectively. For this purpose, we construct a synthetic QDSFM problem (4) by fixing $N = 100$, $R = 100$, and $W_{ii} = 1$, for all $i \in [N]$. We generate each incidence set S_r by choosing uniformly at random a subset of $[N]$ of cardinality 10, and then set the entries of a to be i.i.d. standard Gaussian.

We consider the following submodular function: for $r \in [R]$, $S \subseteq S_r$,

$$F_r(S) = \frac{\min\{|S|, |S_r/S|\}^\theta}{(|S_r|/2)^\theta}, \qquad \theta \in \{0.25, 0.5, 1\}. \tag{13}$$

The number of RCD iterations is set to $300R = 3 \times 10^4$. For the projections performed in the inner loop, as only the conic MNP method is guaranteed to find the exact solution in finite time, we constrain the number of iterations used by competing methods to ensure a fair comparison. To this end, note that for the AMP and PWMP methods (Locatello et al., 2017), a parameter δ that controls the precision is computed as $-\gamma \|\mathbf{d}_t\|^2$ (here we use the notation from Algorithm 3 of (Locatello et al., 2017)). For all algorithms, we terminate the inner loop by setting $\delta = 10^{-12}$ or a maximum number of iterations $|S_r|^3 = 1000$. All algorithms were executed on a 3.2GHz Intel Core i5 machine. For each algorithm, the average and variance of the primal gap are obtained via 100 independent tests. The convergence results are shown in Figure 1, which demonstrates that QRCD-MNP outperforms the other three methods. The performance of QRCD-PWMP is satisfactory when $\theta = 1$, but it deteriorates for smaller values of θ, which may be a consequence of increasing the number of extreme points of cones/polytopes corresponding to the submodular functions (13). QRCD-FW and QRCD-AMP exhibit similar convergence behaviors but fail to provide high-precision solutions for reasonable CPU times.
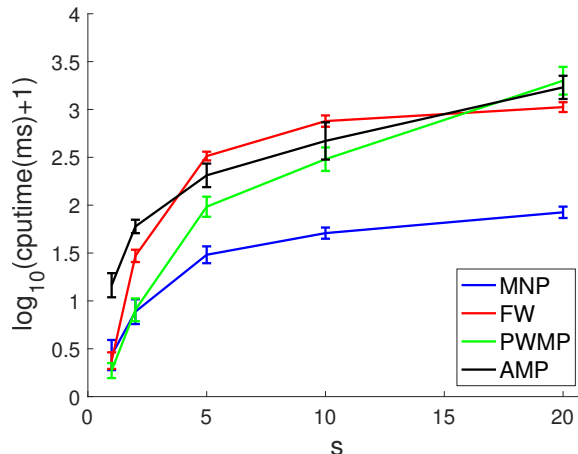
14

Figure 2: The required CPU time for different methods used to compute the projection (12) versus the scale of the submodular function s.

4.5 Numerical Illustration II: Dependency on the Scale of the Submodular Functions

As shown in Theorems 10 and 11, the complexity of the conic MNP/FW methods depends on the scale of the submodular functions F, specifically, $(h(y^{(k)}, \phi^{(k)}) - h^*)/\|a\|_{\tilde{W}}^2 \sim O(F^2)$. We would like to evaluate such dependency. Since we are only concerned with the computations of projection, we focus on a single submodular function $F(\cdot)$ that once again takes the form (13) with parameter $\theta = 0.25$ and a ground set of cardinality $N = |S_r| = 100$. The submodular function is further scaled as $s * F(\cdot)$, where s is a constant in the set $\{1, 2, 5, 10, 20\}$. We generate the entries of a in (12) at the same scale as the submodular function, i.e., $a_v \sim \mathcal{N}(0, s^2)$. The proposed MNP, FW, and the AMP/PWMP methods from (Locatello et al., 2017) were tested, and their CPU execution time reported in Figure 2. All methods terminate when $(h(y^{(k)}, \phi^{(k)}) - h^*)/\|a\|_{\tilde{W}}^2 \leq 10^{-5}$ over independent runs.

As can be seen from Figure 2, the complexities of all methods increase as the submodular function scales up, while the conic MNP method tends to be the most robust with respect to scale changes. PWMP appears to be the most sensitive to scaling due to the growing size of the active set with increasing s.

5. Applications to PageRank

In what follows, we introduce one important application of the QDSFM problem - PageRank (PR). Our treatment of the PR process relies on diffusion processes over hypergraphs (Chan et al., 2017, 2018). We demonstrate that what is known as a *PR vector* can be efficiently computed using our proposed QDSFM solver. We also show that the newly introduced PR retains important properties of the standard PR over graphs, pertaining both to mixing and local partitioning properties (Andersen et al., 2006).

15

5.1 Background: PageRank(PR)

PR is a well-known learning algorithm over graphs, first introduced in the context of web-page search (Page et al., 1999). Given a graph $G = (V, E)$, with $V = [N]$ and E denoting the edge set over V, we let A and D denote the adjacency matrix and diagonal degree matrix of G, respectively. PR essentially reduces to finding a fixed point $p \in \mathbb{R}^N$ of the iterative process

$$p^{(t+1)} = \alpha p^{(0)} + (1 - \alpha) A D^{-1} p^{(t)},$$

where $p^{(0)} \in \mathbb{R}^N$ is a fixed vector and $\alpha \in (0, 1)$. By using different exploration vectors $p^{(0)}$, PR can be adapted to different graph-based applications. For example, for webpages ranking (Page et al., 1999), $p^{(0)}$ may be a uniformly at random chosen vertex. For finding a local graph partitioning around a vertex i (Andersen et al., 2006), $p^{(0)}$ is typically set to e_i, i.e., a zero-one vector with a single 1 in its i-th component.

It is easy to verify that the fixed point p is a solution to the problem

$$\min_p \frac{\alpha}{1 - \alpha} \|p - p^{(0)}\|_{D^{-1}}^2 + (D^{-1}p)^T (D - A)(D^{-1}p) = \|x - a\|_W^2 + \langle x, L(x) \rangle, \qquad (14)$$

where $x = D^{-1}p$, $a = D^{-1}p^{(0)}$, $W = \frac{\alpha}{1-\alpha}D$ and $L = D - A$.

Recent work exploring the role of high-order functional relations among vertices in networks (Benson et al., 2016; Li and Milenkovic, 2017; Li et al., 2017) brought forward the need for studying PR-type of problems for hypergraphs (Zhou et al., 2017; Yin et al., 2017).[1] There exist several definitions of the PR algorithm for hypergraphs. For example, the approach in (Zhou et al., 2017) relies on what is termed a *multilinear PR* (Gleich et al., 2015) and is limited to the setting of uniform hypergraphs, in which all hyperedges have the same cardinality. Computing a multilinear PR requires tensor multiplication and the latter method has complexity exponential in the size of the hyperedges. The PR formulation in (Yin et al., 2017) requires one to first *project hypergraphs onto graphs* and then run standard graph-based PR. The projection step usually causes large distortions when the hyperedges are large and leads to low-quality hypergraph partitionings (Hein et al., 2013; Li and Milenkovic, 2018b).

Using equation (14), we introduce a new formulation for PR on hypergraphs that leverages nonlinear Laplacian operators $L(\cdot)$ for (un)directed hypergraphs (Chan et al., 2017, 2018) and submodular hypergraphs (Li and Milenkovic, 2018b; Yoshida, 2019). The new PR may be applied to both uniform and non-uniform hypergraphs with arbitrarily large hyperedges. The stationary point is the solution to a QDSFM problem, where the term $\langle x, L(x) \rangle$ in (14) is represented by a sum of Lovász extensions of submodular functions defined over hyperedges (See Table 1).

To demonstrate the utility of this particular form of PageRank, we show that the PR vector obtained as a solution of the QDSFM can be used to find a cut of a directed hypergraph (as well as undirected hypergraph) with small conductance, which is an important invariant for local partitioning and community detection (Zhou et al., 2017; Yin et al., 2017). The results essentially generalize the mixing results for PR over graphs derived in (Andersen et al., 2006) and (Andersen et al., 2008). Our main technical contribution is handling new

---

1. Hypergraphs are natural extensions of graphs in which edges are replaced by hyperedges, representing subsets of vertices in V of size ≥ 2.

A single component of $\langle x, L(x) \rangle$	The combinatorial structure	The submodular function
$w_r(x_i - x_j)^2$, $S_r = \{i, j\}$	A graph (Zhou et al., 2004; Zhu et al., 2003b)	$F_r(S) = \sqrt{w_{ij}}$ if $|S \cap \{i, j\}| = 1$
$w_r \max_{i,j \in S_r}(x_i - x_j)^2$	A hypergraph (Hein et al., 2013)	$F_r(S) = \sqrt{w_r}$ if $|S \cap S_r| \in [1, |S_r| - 1]$
$w_r \max_{(i,j) \in H_r \times T_r}(x_i - x_j)_+^2$	A directed hypergraph (Zhang et al., 2017)	$F_r(S) = \sqrt{w_r}$ if $|S \cap H_r| \geq 1$, $|([N]/S) \cap T_r| \geq 1$
General $[f_r(x)]^2$	A submodular hypergraph (Li and Milenkovic, 2018b; Yoshida, 2019)	Any symmetric submodular function

Table 1: The term $\langle x, L(x) \rangle$ for different combinatorial structures. In the third column, whenever the stated conditions are not satisfied, it is assumed that $F_r = 0$. For directed hypergraphs, H_r and T_r are subsets of S_r which we subsequently term the *head* and the *tail set*. For $H_r = T_r = S_r$, one recovers the setting for undirected hypergraphs.

technical challenges that arise from the non-linearity of the operators involved. In what follows, we mainly focus on directed hypergraphs. Tighter and more general results may be obtained for submodular hypergraphs (Li and Milenkovic, 2018b; Yoshida, 2019), which will be reported in a companion paper.

5.2 Terminology: Boundary, Volume, Conductance

Let $G = (V, E)$ be a hypergraph with $V = [N]$ and hyperedge in E equated with the incidence sets $\{S_r\}_{r \in [R]}$. More precisely, each hyperedge represents an incidence set S_r associated with a triplet (w_r, H_r, T_r), where w_r is a scalar weight and H_r and $T_r \subseteq S_r$ are the head and tail sets defined in Table 1. The pair (H_r, T_r) defines a submodular function F_r with incidence set S_r according to $F_r(S) = 1$ if $|S \cap H_r| \geq 1$ and $|\bar{S} \cap T_r| \geq 1$ (where $\bar{S}$ stands for the complement of S), and $F_r(S) = 0$ otherwise. The corresponding Lovász extension reads as $f_r(x) = \max_{(i,j) \in H_r \times T_r}(x_i - x_j)_+$. If S_r contains only two vertices for all values $r \in [R]$, the hypergraph clearly reduces to a graph; and similarly, if $H_r = T_r = S_r$, directed hypergraphs reduce to undirected hypergraphs. We also define the degree of a vertex i as

$$d_i = \sum_{r:i \in S_r} w_r,$$

and let D denote the diagonal matrix of degree values. The volume of a set $S \subseteq V$ is defined as

$$\text{vol}(S) = \sum_{i \in S} d_i.$$

Furthermore, setting $m = \sum_{i \in V} d_i = \text{vol}(V)$ we define the *boundary* of a set S and its *volume* as

$$\partial S = \{r \in [R] | S \cap H_r \neq \emptyset, \bar{S} \cap T_r \neq \emptyset\}, \quad \text{vol}(\partial S) = \sum_{r \in \partial S} w_r = \sum_{r \in [R]} w_r F_r(S).$$

We also define a distribution π_S over V according to $(\pi_S)_i = \frac{d_i}{\text{vol}(S)}$. Furthermore, using the boundary and volume, we also define the *conductance* of the set S as

$$\Phi(S) = \frac{\text{vol}(\partial S)}{\min\{\text{vol}(S), \text{vol}(\bar{S})\}}. \tag{15}$$

The boundary, volume and conductance are all standard and well-studied graph-theoretic concepts (Bollobás, 2013), generalized above to accommodate hypergraphs. The *hypergraph conductance* is defined as the minimal value of the conductance over all possible subsets of V, that is,

$$\Phi^* = \min_{S \subset V, S \neq \emptyset, V} \Phi(S). \tag{16}$$

The hypergraph conductance is an important objective function used for detecting communities of good quality in networks with higher-order functional units (Benson et al., 2016). Solving the optimization (16) naturally yields a two-way partition of V so that the corresponding procedure is also referred to as *hypergraph partitioning*.

5.3 The PageRank Process as QDSFM Optimization

Next, we formally define the PR process for directed hypergraphs and show how it relates to the QDSFM problem. We borrow the definition of a diffusion process (DP) over directed hypergraphs proposed in (Chan et al., 2017, 2018) to describe our PageRank (PR) process. To this end, we first need to define a Markov operator as follows.

Suppose that $p \in \mathbb{R}^N$ is a potential over $[N]$ and let $x = D^{-1}p$. For all hyperedges S_r, we let $S_r^{\downarrow}(x) = \arg\min_{i \in T_r} x_i$ and $S_r^{\uparrow}(x) = \arg\max_{j \in H_r} x_j$. A hyperedge S_r is said to be active if $\arg\max_{j \in H_r} x_j > \arg\min_{i \in T_r} x_i$, and inactive otherwise.

1. For each active hyperedge S_r and for each pair of vertices $(i, j) \in S_r^{\uparrow}(x) \times S_r^{\downarrow}(x)$, we introduce a non-negative scalar $a_r^{(ij)}$ such that $\sum_{(i,j) \in S_r^{\uparrow}(x) \times S_r^{\downarrow}(x)} a_r^{(ij)} = w_r$. For pairs of vertices $(i, j) \notin S_r^{\uparrow}(x) \times S_r^{\downarrow}(x)$ or belonging to inactive hyperedges, we set $a_r^{(ij)} = 0$.

2. Using the above described pairwise weights, we create a symmetric matrix A such that $A_{ij} = \sum_{r \in [R]} a_r^{(ij)}$ and $A_{ii} = d_i - \sum_{i,j \in V, j \neq i} A_{ij}$.

For undirected graphs for which $|S_r| = |T_r| = |H_r| = 2$ for all $r \in [R]$, the matrix A is an adjacency matrix of a graph. Using the terminology of (Chan et al., 2018), we define the Markov operator $M(p)$ based on A according to

$$M(p) = Ax = AD^{-1}p.$$

The Markov operator $M(p)$ is closely related to the Lovász extension f_r, which may be seen as follows. Let $\nabla f_r(x)$ be the subdifferential set of f_r at x. We once again recall that as F_r is submodular, $f_r(x) = \arg\max_{y \in B_r} \langle y, x \rangle$ (Bach, 2013). This gives rise to the following result.

Lemma 12 *For any $p \in \mathbb{R}^N$ and $x = D^{-1}p$, it holds that $p - M(p) \in \sum_{r \in [R]} w_r f_r(x) \nabla f_r(x)$.*

Proof Consider a vertex $i \in V$. Then,

$$(p - M(p))_i = p_i - A_{ii}\frac{p_i}{d_i} - \sum_{j: \in V, j \neq i} A_{ij}\frac{p_j}{d_j} = \sum_{j: \in V, j \neq i} A_{ij}(x_i - x_j) = \sum_{r \in [R]} \sum_{j: \in V, j \neq i} a_r^{(ij)}(x_i - x_j)$$

$$\overset{1)}{=} \sum_{r: \in [R], \, S_r \text{ active}, \, i \in S_r^{\uparrow}} w_r f_r(x) \sum_{j: \in S_r^{\downarrow}} a_r^{(ij)}/w_r + \sum_{r: \in [R], \, S_r \text{ active}, \, i \in S_r^{\downarrow}} w_r f_r(x) \sum_{j: \in S_r^{\uparrow}} (-a_r^{(ij)})/w_r$$

where 1) holds due to the definitions of $a_r^{(ij)}$ and $f_r(x)$. It only remains to show that for a $r \in [R]$, active S_r, the vector v

$$v_i = \begin{cases} \sum_{j: \in S_r^{\downarrow}} a_r^{(ij)}/w_r & i \in S_r^{\uparrow}, \\ -\sum_{j: \in S_r^{\uparrow}} a_r^{(ij)}/w_r & i \in S_r^{\downarrow}, \\ 0 & \text{otherwise} \end{cases}$$

satisfies $v \in \nabla f_r(x)$. Since $\sum_{(i,j) \in S_r^{\uparrow} \times S_r^{\downarrow}} a_r^{(ij)} = w_r$ and $a_r^{(ij)} \geq 0$, we know that $v \in B_r$. Moreover, because $v \in \nabla f_r(x)$, we have $\langle v, x \rangle = f_r(x)$ and thus the claim clearly holds. ∎

The statement of Lemma 12 describes a definition of a Laplacian operator for submodular hypergraphs consistent with previous definitions (Li and Milenkovic, 2018b). We write the underlying Laplacian operator as $L(x) = (D - A)x = p - M(p)$. Note that when $p = \pi_V$, the components of $x = D^{-1}p$ are equal and hence $M(p) = AD^{-1}p = \frac{1}{\text{vol}(S)}A\mathbf{1} = \frac{1}{\text{vol}(S)}D\mathbf{1} = p$. Therefore, π_V may be viewed as an analogue of a stationary distribution.

Given an initial potential $p_0 \in \mathbb{R}^N$ and an $\alpha \in (0, 1]$, the PR process is described by the following ordinary differential equation:

$$\textbf{The PR process} \quad \frac{dp_t}{dt} = \alpha(p_0 - p_t) + (1 - \alpha)(M(p_t) - p_t). \tag{17}$$

The choice of $\alpha = 0$ reduces the PR process to the DP defined in (Chan et al., 2017, 2018). Tracking the PR process at every time point t, i.e., specifying $a_r^{(ij)}$, is as difficult as tracking DP, which in turn requires solving a densest subset problem[2] (Chan et al., 2017, 2018). However, we only need to examine and evaluate the stationary point which corresponds to the PR vector. The PR vector $pr(\alpha, p_0)$ satisfies the equation

$$\alpha(p_0 - pr(\alpha, p_0)) + (1 - \alpha)(M(pr(\alpha, p_0)) - pr(\alpha, p_0)) = 0. \tag{18}$$

---

2. To track the diffusion of $p(t)$ (17), one has to be careful in terms of deciding which components of $p(t)$ are allowed to change within an infinitesimal time. The selection procedure should guarantee that after an infinitesimal time $p(t) \to p(t) + dp(t)$, $M(p(t))$ and $M(p(t) + dp(t))$ satisfy $M(p(t)) = M'p(t)$ and $M(p(t) + dp(t)) = M'p(t) + M'dp(t)$ for some shared choice of the matrix M'. Here, M' captures the active diffusion network that remains unchanged in $[t, t + dt)$. In general, M' will transition through discrete states in a continuous-time domain. Determining which components of $p(t)$ are allowed to change is equivalent to solving a densest set problem.

From Lemma 12, we know that $pr(\alpha, p_0)$ can be obtained by solving a QDSFM. Let $x_{pr} = D^{-1}pr(\alpha, p_0)$. Then

$$\alpha(p_0 - Dx_{pr}) - (1 - \alpha) \sum_{r \in [R]} w_r f_r(x_{pr}) \nabla f_r(x_{pr}) \ni 0$$

$$\Leftrightarrow x_{pr} = \arg\min_x \|x - x_0\|_W^2 + \sum_{r \in [R]} [f_r'(x)]^2, \tag{19}$$

where $x_0 = D^{-1}p_0$, $W = \frac{\alpha}{1-\alpha}D$ and $f_r' = \sqrt{w_r}f_r$.

5.4 Complexity of Computing PageRank over Directed Hypergraphs

We now analyze the computation complexity of computing PageRank over directed hypergraphs using QSDFM solvers. First, due to the specific structure of the Lovász extension f_r' in (19), we devise an algorithm of complexity $O(|S_r| \log |S_r|)$ that *exactly* computes the required projections onto the induced cones. This projection algorithm is much more efficient than the conic FW and MNP methods designed for general polytopes. We focus on projections with respect to one particular value r; since the scaling w_r does not affect the analysis, we henceforth omit the subscript r and the superscript $'$ and simply use f instead of f_r'.

The key idea is to revert the projection $\Pi_C(a; \tilde{W})$ back to its primal form. In the primal setting, optimization becomes straightforward and only requires careful evaluation of the gradient values. First, following a similar strategy as described in Lemma 1, it is easy to show that

$$\min_z \frac{1}{2}\|z - b\|_W^2 + \frac{1}{2}[f(z)]^2$$

is the dual of the problem (12), where $W = \tilde{W}^{-1}$, $b = \frac{1}{2}W^{-1}a$, and f is the Lovász extension corresponding to the base polytope B. Then, one has $y = a - 2Wz$, $\phi = 2\langle y, z \rangle_{W^{-1}}$.

Next, recall that for a directed hyperedge, we introduced head and tail sets H, T, respectively, and $f(z) = (\max_{i \in H} z_j - \min_{j \in T} z_j)_+$. Clearly, when H, T both equal to the corresponding incidence set, the directed hypergraph reduces to an undirected hypergraph. To solve this optimization problem, define two intermediate variables $\gamma = \max_{i \in H} z_i$ and $\delta = \min_{j \in T} z_j$. Denote the derivatives with respect to γ and δ as $\triangle_\gamma$ and $\triangle_\delta$, respectively. Hence, $\triangle_\gamma$ and $\triangle_\delta$ saitsfy

$$\triangle_\gamma = (\gamma - \delta)_+ + \sum_{i \in S_H(\gamma)} W_{ii}(\gamma - b_i), \quad \triangle_\delta = -(\gamma - \delta)_+ + \sum_{j \in S_T(\delta)} W_{jj}(\delta - b_j),$$

where $S_H(\gamma) = \{i | i \in H, b_i \geq \gamma\}$ and $S_T(\delta) = \{j | j \in T, b_j \leq \delta\}$. The optimal values of γ and δ are required to simultaneously satisfy $\triangle_\gamma = 0$ and $\triangle_\delta = 0$. Algorithm 5 is designed to find such values of γ and δ. The "search" for (γ, δ) starts from $(\max_{i \in H} b_i, \min_{j \in T} b_j)$, and one gradually decreases γ and increases δ while ensuring $\triangle_\gamma = -\triangle_\delta$ (see Steps 5-10 in Algorithm 5). The reason for imposing the constraint $\triangle_\gamma = -\triangle_\delta$ during the procedure is to decrease the number of tuning parameters. As a result, the constraint implicitly transforms the procedure into simple line search. The complexity of Algorithm 5 is dominated by the

---

Algorithm 5:   An Exact Projection Algorithm for a Directed Hyperedge

---

Input: W, b, H, T

1. Sort $\{b_i\}_{i \in H}$ and $\{b_j\}_{j \in T}$.
2. Initialize $\gamma \leftarrow \max_{i \in H} b_i$ and $\delta \leftarrow \min_{j \in T} b_j$.
3. **If** $\gamma \leq \delta$, **return** $z = b$.
4. Iteratively execute:
5. $w_H \leftarrow \sum_{i \in S_H(\gamma)} W_{ii}$, $w_T \leftarrow \sum_{j \in S_T(\delta)} W_{jj}$
6. $\gamma_1 \leftarrow \max_{i \in H/S_H(\gamma)} b_v$, $\delta_1 \leftarrow \delta + (\gamma - \gamma_1) w_H / w_T$
7. $\delta_2 \leftarrow \min_{j \in T/S_T(\delta)} b_v$, $\gamma_2 \leftarrow \gamma - (\delta_2 - \delta) w_T / w_H$
8. $k^* \leftarrow \arg \min_{k \in \{1,2\}} \delta_k$
9.   **If** $\gamma_{k^*} \leq \delta_{k^*}$ or $\triangle_{\gamma_{k^*}} \leq 0$, **break**
10. $(\gamma, \delta) \leftarrow (\gamma_{k^*}, \delta_{k^*})$
11. $(\gamma, \delta) \leftarrow (w_T, w_H) \frac{\triangle_\gamma}{w_T w_H + w_T + w_H}$
12. Set z_i to γ, if $i \in S_H(\gamma)$, to δ, if $i \in S_T(\delta)$, and to b_i otherwise.

---

sorting step (Step 1), which requires $O(|S_r| \log |S_r|)$ operations, as w_H, w_T, $\triangle_\gamma$, and $\triangle_\delta$ can all be efficiently tracked within the inner loops.

Combining the projection method in Algorithm 5 and Corollary 9 with parameter choice $\beta = \frac{\alpha}{1-\alpha}$, we arrive at the following result summarizing the overall complexity of running the PageRank process on (un)directed hypergraphs.

Corollary 13 *The PageRank problem on (un)directed hypergraphs can be obtained by solving (19). A combination of Algorithm 1 (RCD) and Algorithm 5 returns an ϵ-optimal solution in expectation with total computation complexity:*

$$O\left(N^2 \max\left\{1, 9\frac{1-\alpha}{\alpha}\right\} \max_{i,j \in [N]} \frac{D_{ii}}{D_{jj}} \sum_{r \in [R]} |S_r| \log |S_r| \log \frac{1}{\epsilon}\right).$$

A combination of Algorithm 2 (AP) and Algorithm 5 returns an ϵ-optimal solution with total computation complexity:

$$O\left(N^2 \max\left\{1, 9\frac{1-\alpha}{\alpha}\right\} \max_{i,j \in [N]} \frac{\Psi_{jj} D_{ii}}{D_{jj}} \sum_{r \in [R]} |S_r| \log |S_r| \log \frac{1}{\epsilon}\right).$$

5.5 Analysis of Partitions Based on PageRank

We are now ready to analyze hypergraph partitioning procedures (16) based on our definition of PageRank. The optimization problem (16) is NP-hard even for graphs so we only seek approximate solutions. Consequently, an important application of our version of PageRank is to provide an approximate solution for (16). We start by proving that our version of PageRank, similarly to the standard PageRank over graphs (Andersen et al., 2006), satisfies a mixing result derived based on so-called Lovász-Simonovits curves (Lovasz and Simonovits, 1990). This result may be further used to prove that *Personalized* PageR-

ank leads to partitions with small conductance[3]. The main components of our proof are inspired by the proofs for PageRank over graphs (Andersen et al., 2006). The novelty of our approach is that we need several specialized steps to handle the nonlinearity of the Markov operator $M(\cdot)$, which in the standard graph setting is linear. We postpone the proofs of all our results to Section 7.7.

We first introduce relevant concepts and terminology necessary for presenting our results.

Sweep cuts are used to partition hypergraphs based on the order of components of a distribution p and are defined as follows. Set $x = D^{-1}p$ and sort the components of x so that

$$x_{i_1} \geq x_{i_2} \geq \cdots \geq x_{i_N}. \tag{20}$$

Let $\mathcal{S}_j^p = \{x_{i_1}, x_{i_2}, ..., x_{i_j}\}$. Recall the definition of the conductance of a set $\Phi(\cdot)$ (15) and evaluate

$$j^* = \arg \min_{j \in [N]} \Phi(\mathcal{S}_j^p).$$

The pair $(\mathcal{S}_{j^*}^p, \bar{\mathcal{S}}_{j^*}^p)$ is referred to as a *sweep cut*. Furthermore, define $\Phi_p = \Phi(\mathcal{S}_{j^*}^p)$; in words, Φ_p is the smallest conductance that may be achieved by sweep cuts of a potential p. Our target is to show that the conductance based on a Personalized PR vector, i.e., $\Phi_{pr(\alpha,1_v)}$, may approximate Φ^*.

Intuitively, the conductance of a hypergraph for a given p_0 and α captures how "close" a PR vector $pr(\alpha, p_0)$ is to the stationary distribution π_V: a large conductance results in mixing the potentials of different vertices and thus makes $pr(\alpha, p_0)$ closer to π_V. To characterize the degree of "closeness", we introduce the Lovász-Simonovits curve. Given a potential vector $p \in \mathbb{R}^N$ and $x = D^{-1}p$, suppose that the order of the components in x follows equation (20). Let $\mathrm{vol}(\mathcal{S}_0^p) = 0$. Define a piecewise linear function $I_p(\cdot) : [0, m] \to [0, 1]$ according to

$$I_p(z) = p(\mathcal{S}_{j-1}^p) + \frac{z - \mathrm{vol}(\mathcal{S}_{j-1}^p)}{d_{v_j}} p_{v_j}, \text{ for } \mathrm{vol}(\mathcal{S}_{j-1}^p) \leq k \leq \mathrm{vol}(\mathcal{S}_j^p), \ j \in [N].$$

Note that the Lovász-Simonovits curve of π_V contains only one segment by its very definition. It is easy to check that $I_p(z)$ is continuous and concave in z. Moreover, for any set $S \subseteq [N]$, we have $p(S) \leq I_p(\mathrm{vol}(S))$. We further write $V_p = \{j : x_{i_j} > x_{i_{j+1}}\}$ and refer to $\{\mathrm{vol}(\mathcal{S}_j^p)\}_{j \in V_p}$ as the (exact) break points of I_p. The relevance of the Lovász-Simonovits curve in the aforementioned context may be understood as follows. Although the Lovász-Simonovits curve is piecewise linear, the changes of the slopes at the break points may be used to describe its "curvatures". A large (small) curvature corresponds to the case that the distribution of the potential is far from (close to) the stationary distribution, which further implies a small (large) conductance of the graphs/hypergraphs (see Theorem 16). Therefore, analyzing the Lovász-Simonovits curvature is of crucial importance in the process of bounding the conductance. The first step of this analysis is presented in Lemma 14, which establishes the "curvatures" at the break points.

---

3. Recall that personalized PageRank is a PR process with initial distribution $p_0 = 1_i$, for some vertex $i \in V$.

Lemma 14 *Let $p = pr(\alpha, p_0)$, $x = D^{-1}p$ and $j \in V_p$. Then,*

$$I_p(vol(\mathcal{S}_j^p)) \le \frac{\alpha}{2-\alpha} p_0(\mathcal{S}_j^p) + \frac{1-\alpha}{2-\alpha} [I_p(vol(\mathcal{S}_j^p) - vol(\partial \mathcal{S}_j^p)) + I_p(vol(\mathcal{S}_j^p) + vol(\partial \mathcal{S}_j^p))].$$

Furthermore, for $k \in [0, m]$,

$$I_p(k) \le p_0(k) \le I_{p_0}(k).$$

Remark 15 *In comparison with the undirected graph case (Lemma 5 (Andersen et al., 2006)), where the result holds for arbitrary $S \subseteq V$, our claim is true only for $\mathcal{S}_j^p$ for which $j \in V_p$. However, this result suffices to establish all relevant PR results.*

Using the upper bound on the break points in I_p, we can now construct a curve that uniformly bounds I_p using Φ_p.

Theorem 16 *Let $p = pr(\alpha, p_0)$ be the previously defined PR vector, and let Φ_p be the minimal conductance achieved by a sweep cut. For any integer $t \ge 0$ and any $k \in [0, m]$, the following bound holds:*

$$I_p(k) \le \frac{k}{m} + \frac{\alpha}{2-\alpha} t + \sqrt{\frac{\min\{k, m-k\}}{\min_{i:(p_0)_i > 0} d_i}} \left(1 - \frac{\Phi_p^2}{8}\right)^t.$$

For graphs (Andersen et al., 2006), Theorem 16 may be used to characterize the graph partitioning property of sweep cuts induced by a personalized PageRank vector. Similarly, for general directed hypergraphs, we establish a similar result in Theorem 17.

Theorem 17 *Let S be a set of vertices such that $vol(S) \le \frac{1}{2}M$ and $\Phi(S) \le \frac{\alpha}{c}$, for some constants α, c. If there exists a distribution P for sampling vertices $v \in S$ such that $\mathbb{E}_{i \sim P}[pr(\alpha, 1_i)(\bar{S})] \le \frac{c}{8} pr(\alpha, \pi_S)(\bar{S})$, then with probability at least $\frac{1}{2}$,*

$$\Phi_{pr(\alpha, 1_i)} = O(\sqrt{\alpha \log \frac{vol(S)}{d_i}}),$$

where i is sampled according to P.

Remark 18 *Note that in Theorem 17, we require that the sampling distribution P satisfy $\mathbb{E}_{i \sim P}[pr(\alpha, 1_i)(\bar{S})] \le \frac{c}{8} pr(\alpha, \pi_S)(\bar{S})$. For graphs, when we sample a vertex i with probability proportional to its degree d_i, this condition is naturally satisfied, with $c = 8$. However, for general (un)directed hypergraphs, the sampling procedure is non-trivial and harder to handle due to the non-linearity of the random-walk operator $M(\cdot)$. We relegate a more in-depth study of this topic to a companion paper.*

6. Applications to Semi-supervised Learning and Numerical Experiments

Another important application of QDSFM is semi-supervised learning (SSL). SSL is a learning paradigm that allows one to utilize the underlying structure or distribution of unlabeled samples whenever the information provided by labeled samples is insufficient for learning an inductive predictor (Gammerman et al., 1998; Joachims, 2003). A standard setup for a K-class transductive learner is as follows: given N data points $\{z_i\}_{i \in [N]}$, along with labels for the first l ($\ll N$) samples $\{y_i | y_i \in [K]\}_{i \in [l]}$, the learner is asked to infer the labels of all the remaining data points $i \in [N]/[l]$. The widely-used SSL problem with least square loss requires one to solve K regularized problems. For each class $k \in [K]$, one sets the scores of data points within the class to

$$\hat{x}^{(k)} = \arg\min_{x^{(k)}} \beta \|x^{(k)} - a^{(k)}\|^2 + \Omega(x^{(k)}),$$

where $a^{(k)}$ describes the information provided by the known labels, i.e., $a_i^{(k)} = 1$ if $y_i = k$, and 0 otherwise, β denotes a hyperparameter and Ω is a smoothness regularizer. The labels of the data points are estimated as

$$\hat{y}_i = \arg\max_k \{\hat{x}_i^{(k)}\}.$$

In typical graph and hypergraph learning problems, Ω is chosen to be a Laplacian regularizer constructed using $\{z_i\}_{i \in [N]}$; this regularization term ties the above learning problems to PageRank (again, refer to Table 1). With the Laplacian regularization, each edge/hyperedge corresponds to one decomposed part f_r in the QDSFM problem. The variables themselves may be normalized with respect to their degrees, in which case the normalized Laplacian is used instead. For example, in graph learning, one of the terms in Ω assumes the form $w_{ij}(x_i/\sqrt{d_i} - x_j/\sqrt{d_j})^2$, where d_i and d_j correspond to the degrees of the vertex variables i and j, respectively. Using some simple algebra, it can be shown that the normalization term is embedded into the matrix W used in the definition of the QDSFM problem (4).

6.1 Experimental Setup

We numerically evaluate our SSL learning framework for hypergraphs on both real and synthetic datasets. For the particular problem at hand, the QDSFM problem can be formulated as follows:

$$\min_{x \in \mathbb{R}^N} \beta \|x - a\|^2 + \sum_{r \in [R]} \max_{i,j \in S_r} \left(\frac{x_i}{\sqrt{W_{ii}}} - \frac{x_j}{\sqrt{W_{jj}}}\right)^2, \tag{21}$$

where $a_i \in \{-1, 0, 1\}$ indicates if the corresponding variable i has a negative, missing, or positive label, respectively, β is a parameter that balances out the influence of observations and the regularization term, $\{W_{ii}\}_{i \in [N]}$ defines a positive measure over the variables and may be chosen to be the degree matrix D, with $D_{ii} = |\{r \in [R] : i \in S_r\}|$. Each part in the decomposition corresponds to one hyperedge. We compare eight different solvers falling into three categories: (a) our proposed general QDSFM solvers, *QRCD-SPE, QRCD-MNP, QRCD-FW and QAP-SPE*; (b) alternative specialized solvers for the given problem (21), including *PDHG* (Hein et al., 2013) and *SGD* (Zhang et al., 2017); (c) SSL solvers that

do not use QDSFM as the objective, including *DRCD* (Ene and Nguyen, 2015) and *InvLap* (Zhou et al., 2007). The first three methods all have outer-loops that execute RCD, but with different inner-loop projections computed via the *exact projection algorithm* for undirected hyperedges (see Algorithm 5 in Section 5.4), or the generic MNP and FW. As it may be time consuming to find the precise projections via MNP and FW, we always fix the number of MAJOR loops of the MNP and the number of iterations of the FW method to $100|S_r|$ and $100|S_r|^2$, respectively. Empirically, these choices provide an acceptable trade-off between accuracy and complexity. The QAP-SPE method uses AP in the outer-loop and exact inner-loop projections (see Algorithm 5 in Section 5.4). PDHG and SGD are the only known solvers for the specific objective (21). PDHG and SGD depend on certain parameters that we choose in standard fashion: for PDHG, we set $\sigma = \tau = \frac{1}{\sqrt{1+\max_i D_{ii}}}$ and for SGD, we set $\eta_k = \frac{1}{k\beta \max_i W_{ii}}$.

DRCD is a state-of-the-art solver for DSFM and also uses a combination of outer-loop RCD and inner-loop projections. InvLap first transforms hyperedges into cliques and then solves a Laplacian-based linear problem. All the aforementioned methods, except InvLap, are implemented in C++ in a nonparallel fashion. InvLap is executed via matrix inversion operations in Matlab, and may be parallelized. The CPU times of all methods are recorded on a 3.2GHz Intel Core i5. The results are summarized for 100 independent tests. When reporting the results, we use the primal gap ("gap") [4] to characterize the convergence properties of different solvers.

6.2 Synthetic Data

We generated a hypergraph with $N = 1000$ vertices that belong to two equal-sized clusters. We uniformly at random generated 500 hyperedges within each cluster and 1000 hyperedges across the two clusters. Note that in higher-order clustering, we do not need to have many hyperedges within each cluster to obtain good clustering results. Each hyperedge includes 20 vertices. We also uniformly at random picked $l = 1, 2, 3, 4$ vertices from each cluster to represent labeled datapoints. With the vector x obtained by solving (21), we classified the variables based on the Cheeger cut rule (Hein et al., 2013): suppose that $\frac{x_{i_1}}{\sqrt{W_{i_1 i_1}}} \geq \frac{x_{i_2}}{\sqrt{W_{i_2 i_2}}} \geq \cdots \geq \frac{x_{i_N}}{\sqrt{W_{i_N i_N}}}$, and let $\mathcal{S}_j = \{i_1, i_2, ..., i_j\}$. We partitioned $[N]$ into two sets $(\mathcal{S}_{j^*}, \bar{\mathcal{S}}_{j^*})$, where

$$j^* = \arg\min_{j \in [N]} \Phi(\mathcal{S}_j) \triangleq \frac{|S_r \cap \mathcal{S}_j \neq \emptyset, S_r \cap \bar{\mathcal{S}}_j \neq \emptyset\}|}{\max\{\sum_{r \in [R]} |S_r \cap \mathcal{S}_j|, \sum_{r \in [R]} |S_r \cap \bar{\mathcal{S}}_j|\}}.$$

The classification error is defined as (# of incorrectly classified vertices)$/N$. In the experiment, we used $W_{ii} = D_{ii}, \forall i$, and tuned β to be nearly optimal for different objectives with respect to the classification error rates: for QDSFM as the objective, using QRCD-SPE, QAP-SPE, PDHG, and SGD as the methods of choice, we set $\beta = 0.02$; for DSFM as the objective, including the DRCD method, we set $\beta = 1$; for InvLap, we set $\beta = 0.001$.

---

4. We place a high accuracy requirement for QDSFM-MNP by ignoring the CPU time needed to guarantee that it achieves a duality gap as low as $1e - 14$. Note that the duality gap is computed according to $\|x^{(k)} - a\|_W^2 + \sum_{r \in [R]} \left[f_r(x^{(k)})\right]^2 - g(y^{(k)}, \phi^{(k)})/4$, where $x^{(k)} = a - \frac{1}{2}W^{-1} \sum_{r \in [R]} y_r^{(k)}$, which is an upper bound on the primal gap. Therefore, the achieved value of the objective has high accuracy and thus can be used as the optimal primal value.
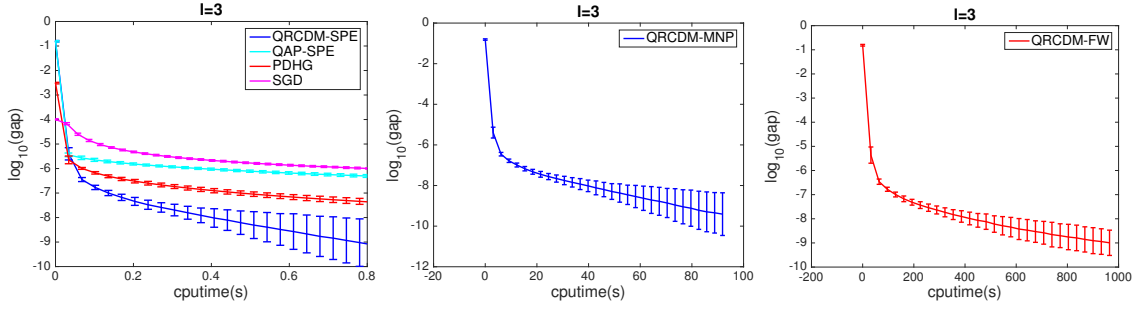
Figure 3: Convergence results on synthetic datasets: gap vs CPU time of different QDSFM solvers (with an average ± standard deviation).

Obj.	Solvers	Classification error rate (%)								Average $100c(\mathcal{S}_{j*})$				#iterations	cputime(s)
		l=1		l=2		l=3		l=4		l=1	l=2	l=3	l=4		
		MN	MD	MN	MD	MN	MD	MN	MD						
QDSFM	QRCD-SPE	**2.93**	**2.55**	**2.23**	**0.00**	**1.47**	**0.00**	**0.78**	**0.00**	6.81	6.04	5.71	5.41	4.8×10^5	0.83
	QAP-SPE	14.9	15.0	12.6	13.2	7.33	8.10	4.07	3.80	9.51	9.21	8.14	7.09	2.7×10^2	0.85
	PDHG	9.05	9.65	4.56	4.05	3.02	2.55	1.74	0.95	8.64	7.32	6.81	6.11	3.0×10^2	0.83
	SGD	5.79	4.15	4.30	3.30	3.94	2.90	3.41	2.10	8.22	7.11	7.01	6.53	1.5×10^4	0.86
Oth.	DRCD	44.7	44.2	46.1	45.3	43.4	44.2	45.3	44.6	9.97	9.97	9.96	9.97	3.8×10^6	0.85
	InvLap	8.17	7.30	3.27	3.00	1.91	1.60	0.89	0.70	8.89	7.11	6.18	5.60	—	**0.07**

Table 2: Prediction results on synthetic datasets: classification error rates & Average 100 $c(\mathcal{S}_{j*})$ for different solvers (MN: mean, MD: median).

The left subfigure of Figure 3 shows that QRCD-SPE converges much faster than all other methods when solving the problem (21) according to the gap metric (we only plotted the curve for $l = 3$ as all other values of l produce similar patterns). To avoid clutter, we moved the results for QRCD-MNP and QRCD-FW to the right two subfigures in Figure 3, as these methods are typically 100 to 1000 times slower than QRCD-SPE. In Table 2, we described the performance of different methods with comparable CPUtimes. Note that when QRCD-SPE converges (with primal gap 10^{-9} achieved after 0.83s), the obtained x leads to a much smaller classification error than other methods. QAP-SPE, PDHG and SGD all have large classification errors as they do not converge within short CPU time-frames. QAP-SPE and PDHG perform only a small number of iterations, but each iteration computes the projections for all the hyperedges, which is more time-consuming. The poor performance of DRCD implies that the DFSM is not a good objective for SSL. InvLap achieves moderate classification errors, but still does not match the performance of QRCD-SPE. Furthermore, note that InvLap uses Matlab, which is optimized for matrix operations, thus fairly efficient. However, for experiments on real datasets with fewer but significantly larger hyperedges, InvLap does not offer as good performance as the one for synthetic data. The column "Average $100c(\mathcal{S}_{j*})$" also illustrates that the QDSFM objective is a good choice for finding approximate balanced cuts of hypergraphs.

We also evaluated the influence of parameter choices on the convergence of QRCD methods. According to Theorem 4, the required number of RCD iterations for achieving an ϵ-optimal solution for (21) is roughly $O(N^2 R \max(1, 9/(2\beta)) \max_{i,j \in [N]} W_{ii}/W_{jj} \log 1/\epsilon)$ (see Section 7.8). We mainly focus on testing the dependence on the parameters β and $\max_{i,j \in [N]} W_{ii}/W_{jj}$, as the term $N^2 R$ is also included in the iteration complexity of DSFM and was shown to be necessary given certain submodular structures (Li and Milenkovic,
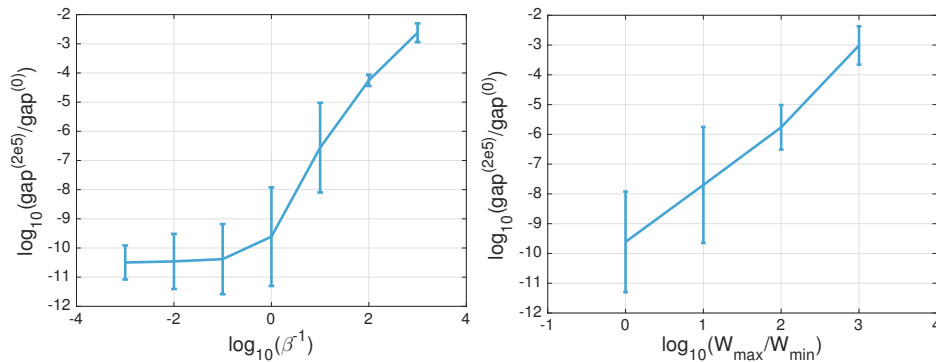
Figure 4: Parameter sensitivity: the rate of a primal gap of QRCD after 2×10^5 iterations with respect to different choices of the parameters β & $\max_{i,j \in [N]} W_{ii}/W_{jj}$.
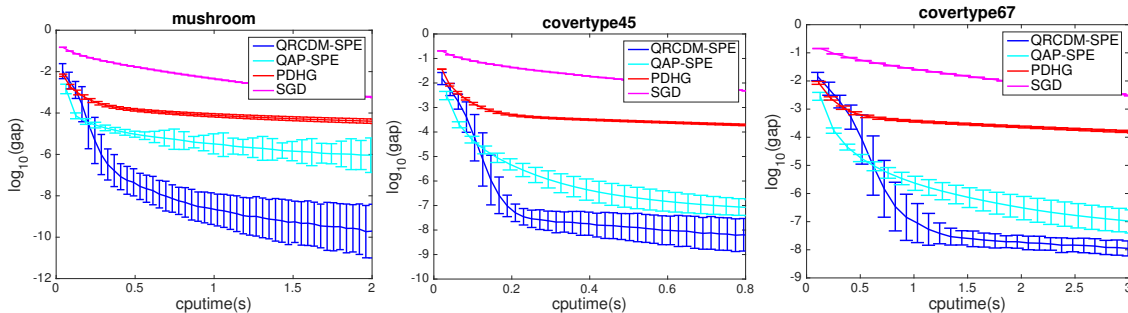

Figure 5: Convergence of different solvers for QDFSM over three different real datasets.

2018a). To test the effect of β, we fix $W_{ii} = 1$ for all i, and vary $\beta \in [10^{-3}, 10^3]$. To test the effect of W, we fix $\beta = 1$ and randomly choose half of the vertices and set their W_{ii} values to lie in $\{1, 0.1, 0.01, 0.001\}$, and set the remaining ones to 1. Figure 4 shows our results. The logarithm of gap ratios is proportional to $\log \beta^{-1}$ for small β, and $\log \max_{i,j \in [N]} W_{ii}/W_{jj}$, which is not as sensitive as predicted by Theorem 4. Moreover, when β is relatively large (> 1), the dependence on β levels out.

6.3 Real Data

We also evaluated the proposed algorithms on three UCI datasets: *Mushroom, Covertype45, Covertype67*, used as standard test datasets for SSL on hypergraphs (Zhou et al., 2007; Hein et al., 2013; Zhang et al., 2017). Each dataset corresponds to a hypergraph model described in (Hein et al., 2013): Entries correspond to vertices, while each categorical feature is modeled as one hyperedge; numerical features are first quantized into 10 bins of equal size, and then mapped to hyperedges. Compared to synthetic data, in the real datasets, the size of most hyperedges is significantly larger (≥ 1000) while the number of hyperedges is small (≈ 100). See Table 3 for a detailed description of the parameters of the generated hypergraphs. Previous work has shown that smaller classification errors can be achieved by using QDSFM as the objective instead of DSFM or InvLap (Hein et al., 2013). In our experiment, we focused on comparing the convergence of different solvers for QDSFM. We

Dataset	Mushroom	Covertype45	Covertype67
N	8124	12240	37877
R	112	127	136
$\sum_{r \in [R]} |S_r|$	170604	145999	451529

Table 3: The UCI datasets used for experimental testing.

set $\beta = 100$ and $W_{ii} = 1$, for all i, and set the number of observed labels to 100, which is a proper setting according to (Hein et al., 2013). Figure 5 shows the results. Again, the proposed QRCD-SPE and QAP-SPE methods both converge faster than PDHG and SGD, while QRCD-SPE performs the best. Note that we did not plot the results for QRCD-MNP and QRCD-FW as these methods converge extremely slowly due to the large sizes of the hyperedges. InvLap requires 22, 114 and 1802 seconds to run on the Mushroom, Covertype45, and Covertype67 datasets, respectively. Hence, the method does not scale well with the problem size.

7. Detailed Proofs

7.1 Proofs of the Results Pertaining to the Dual Formulation

7.1.1 Proof of Lemma 1

In all derivations that follow, we may exchange the order of minimization and maximization (i.e., $\min \max = \max \min$) due to Proposition 2.2 (Ekeland and Temam, 1999). Plugging equation (5) into (4), we obtain

$$\min_x \sum_{r \in [R]} [f_r(x)]^2 + \|x - a\|_W^2$$

$$= \min_x \max_{\phi_r \geq 0, y_r \in \phi_r B_r} \sum_{r \in [R]} \left[\langle y_r, x \rangle - \frac{\phi_r^2}{4} \right] + \|x - a\|_W^2$$

$$= \max_{\phi_r \geq 0, y_r \in \phi_r B_r} \min_x \sum_{r \in [R]} \left[\langle y_r, x \rangle - \frac{\phi_r^2}{4} \right] + \|x - a\|_W^2$$

$$= \max_{\phi_r \geq 0, y_r \in \phi_r B_r} -\frac{1}{4} \| \sum_{r \in [R]} y_r - 2Wa \|_{W^{-1}}^2 - \frac{1}{4} \sum_r \phi_r^2 + \|a\|_W^2.$$

By eliminating some constants, one obtains the dual (6).

Next, we prove that the problem (7) is equivalent to (6) when removing λ_r.

28

First, (7) is equivalent to

$$\min_{\phi_r \geq 0, y_r \in \phi_r B_r, \lambda_r} \max_\lambda \sum_{r \in [R]} \left[\|y_r - \frac{\lambda_r}{\sqrt{R}}\|_{W^{-1}}^2 + \phi_r^2 \right] + \left\langle \lambda, \sum_{r \in [R]} \lambda_r - 2Wa \right\rangle$$

$$= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \max_\lambda \min_{\lambda_r} \sum_{r \in [R]} \left[\|y_r - \frac{\lambda_r}{\sqrt{R}}\|_{W^{-1}}^2 + \phi_r^2 \right] + \left\langle \lambda, \sum_{r \in [R]} \lambda_r - 2Wa \right\rangle$$

$$= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \max_\lambda \sum_{r \in [R]} \left[\frac{1}{4} \|\lambda\|_W^2 + \phi_r^2 \right] + \left\langle \lambda, \sqrt{R} \sum_{r \in [R]} (y_r - \frac{1}{2} W \lambda_r) - 2Wa \right\rangle$$

$$= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \max_\lambda -\frac{R}{4} \|\lambda\|_W^2 + \sqrt{R} \left\langle \lambda, \sum_{r \in [R]} y_r - 2Wa \right\rangle + \sum_{r \in [R]} \phi_r^2$$

$$= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \| \sum_{r \in [R]} y_r - 2Wa \|_{W^{-1}}^2 + \sum_{r \in [R]} \phi_r^2,$$

which is equivalent to (6).

7.1.2 PROOF OF LEMMA 2

We start by recalling the following lemma from (Li and Milenkovic, 2018a) that characterizes the geometric structure of the product base polytope.

Lemma 19 (Li and Milenkovic (2018a)) *Assume that $W \in \mathbb{R}^{N \times N}$ is a positive diagonal matrix. Let $y \in \bigotimes_{r \in [R]} \phi_r' B_r$ and let s be in the base polytope of the submodular function $\sum_r \phi_r' F_r$. Then, there exists a point $y' \in \bigotimes_{r \in [R]} \phi_r' B_r$ such that $\sum_{r \in [R]} y_r' = s$ and*

$$\|y' - y\|_{I(W)} \leq \sqrt{\frac{\sum_{i=1}^N W_{ii}}{2}} \| \sum_{r \in [R]} y_r - s \|_1.$$

Lemma 19 cannot be used directly to prove Lemma 2, since y, y' are in different product base polytopes, $\bigotimes_{r \in [R]} \phi_r B_r$ and $\bigotimes_{r \in [R]} \phi_r' B_r$, respectively. However, the following lemma shows that one can transform y to lie in the same base polytopes that contains y'.

Lemma 20 *For a given feasible point $(y, \phi) \in \bigotimes_{r \in [R]} C_r$, and a nonnegative vector $\phi' = (\phi_r) \in \bigotimes_{r \in [R]} \mathbb{R}_{\geq 0}$, one has*

$$\| \sum_{r \in [R]} y_r - s \|_1 + \frac{\rho}{2} \|\phi' - \phi\| \geq \| \sum_{r \in [R]} \frac{\phi_r'}{\phi_r} y_r - s \|_1.$$

Proof For all r, let $\tilde{y}_r = y_r/\phi_r \in B_r$, and define a function that depends on ϕ,

$$h(\phi) = \| \sum_{r \in [R]} y_r - s \|_1 = \| \sum_{r \in [R]} \phi_r \tilde{y}_r - s \|_1.$$

29

For all ϕ and r, $|\nabla_{\phi_r} h(\phi)| \leq \|\tilde{y}_r\|_1$. Therefore,

$$
\begin{aligned}
h(\phi') &= h(\phi) + \int_{t=0}^{1} \langle \nabla h|_{\phi+t(\phi'-\phi)}, t(\phi'-\phi) \rangle dt \\
&\geq h(\phi) - \frac{\max_{t \in [0,1]} \|\nabla h|_{\phi+t(\phi'-\phi)}\|}{2} \|\phi' - \phi\| \\
&\geq h(\phi) - \frac{\max_{\tilde{y}_r \in B_r, \forall r} \sqrt{\sum_{r \in [R]} \|\tilde{y}_r\|_1^2}}{2} \|\phi' - \phi\| = h(\phi) - \frac{\rho}{2} \|\phi' - \phi\|.
\end{aligned}
$$

∎

Combining Lemma 20 with Lemma 19, we can establish the claim of Lemma 2.

First, let $\rho(W^{(1)}) = \max_{y \in \bigotimes_{r \in [R]} B_r} \sqrt{\sum_{r \in [R]} \|y_r\|_{W_r}^2}$. Suppose that $y' \in \bigotimes_{r \in [R]} \phi'_r B_r$ is such that $\sum_{r \in [R]} y'_r = s$ and it minimizes $\sum_{r \in [R]} \|\frac{\phi'_r}{\phi_r} y_r - y'_r\|_{W^{(1)}}^2$. As s lies in the base polytope of $\sum_{r \in [R]} \phi'_r F_r$, we know that such an y' exists. Moreover, we have

$$
\|y - y'\|_{I(W^{(1)})} \leq \sum_{r \in [R]} \|y'_r - \frac{\phi'_r}{\phi_r} y_r\|_{W^{(1)}} + \sum_{r \in [R]} \|y_r - \frac{\phi'_r}{\phi_r} y_r\|_{W^{(1)}}
$$

$$
\overset{1)}{\leq} \sqrt{\frac{\sum_{i \in [N]} W_{ii}^{(1)}}{2}} \| \sum_{r \in [R]} \frac{\phi'_r}{\phi_r} y_r - s\|_1 + \rho(W^{(1)}) \|\phi' - \phi\|
$$

$$
\overset{2)}{\leq} \sqrt{\frac{\sum_{i \in [N]} W_{ii}^{(1)}}{2}} \left[\| \sum_{r \in [R]} y_r - s\|_1 + \frac{\rho}{2} \|\phi' - \phi\| \right] + \rho(W^{(1)}) \|\phi' - \phi\|
$$

$$
= \sqrt{\frac{\sum_{i \in [N]} W_{ii}^{(1)} \sum_{j \in [N]} 1/W_{jj}^{(2)}}{2}} \| \sum_{r \in [R]} y_r - s\|_{W^{(2)}}
$$

$$
+ \left[\sqrt{\frac{\sum_{i \in [N]} W_{ii}^{(1)}}{2}} \frac{\rho}{2} + \rho(W^{(1)}) \right] \|\phi' - \phi\|
$$

$$
\overset{3)}{\leq} \sqrt{\frac{\sum_{i \in [N]} W_{ii}^{(1)} \sum_{j \in [N]} 1/W_{jj}^{(2)}}{2}} \| \sum_{r \in [R]} y_r - a\|_{W^{(2)}} + \frac{3}{2} \sqrt{\frac{\sum_{i \in [N]} W_{ii}^{(1)}}{2}} \rho \|\phi' - \phi\|,
$$

where 1) follows from Lemma 19 and the definition of $\rho(W^{(1)})$, 2) is a consequence of Lemma 20 and 3) holds because

$$
\sum_{i \in [N]} W_{ii}^{(1)} \sum_{r \in [R]} \|y_r\|_1^2 \geq \sum_{r \in [R]} \|y_r\|_{W_r}^2.
$$

7.2 Proof for the Linear Convergence Rate of the RCD Algorithm

7.2.1 PROOF OF LEMMA 3

If (y^*, ϕ^*) is the optimal solution, then it must hold that $\sum_{r \in [R]} y_r^* = 2W(a - x^*)$ because of the duality between the primal and dual variables. Moreover, we also know that there must exist a nonempty collection $\mathcal{Y}$ of points $y' \in \bigotimes_{r \in [R]} \phi_r^* B_r$ such that $\sum_{r \in [R]} y_r' = \sum_{r \in [R]} y_r^*$. Using Lemma 2, and setting $\phi' = \phi^*, s = 2W(a - x^*)$, $W^{(1)}, W^{(2)} = W^{-1}$, we can show that there exists some $y' \in \mathcal{Y}$ such that

$$\|y - y'\|_{I(W^{-1})}^2 + \|\phi - \phi^*\|^2 \leq \mu(W^{-1}, W^{-1}) \left[\| \sum_{r \in [R]} (y_r - y_r')\|_{W^{-1}}^2 + \|\phi - \phi^*\|^2 \right].$$

According to the definition of y^*, one has $\|y - y^*\|_{I(W^{-1})}^2 \leq \|y - y'\|_{I(W^{-1})}^2$ for $y' \in \mathcal{Y}$. This concludes the proof.

7.2.2 PROOF OF THEOREM 4

First, suppose that $(y^*, \phi^*) = \arg\min_{(y', \phi') \in \Xi} \|y^{(k)} - y'\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi'\|^2$. Throughout the proof, for simplicity, we use μ to denote $\mu(W^{-1}, W^{-1})$. We start by establishing the following results.

Lemma 21 *It can be shown that the following inequalities hold:*

$$\langle \nabla g(y^{(k)}, \phi^{(k)}), (y^* - y^{(k)}, \phi^* - \phi^{(k)}) \rangle$$

$$\overset{1)}{\leq} g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)}) - \frac{1}{\mu} \left(\|y^{(k)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi^*\|^2 \right)$$

$$\overset{2)}{\leq} \frac{2}{\mu + 1} \left[g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)}) - \|y^{(k)} - y^*\|_{I(W^{-1})}^2 - \|\phi^{(k)} - \phi^*\|^2 \right]. \tag{22}$$

Proof From Lemma 3, we can infer that

$$\| \sum_{r \in [R]} (y_r - y_r^*)\|_{W^{-1}}^2 + \|\phi - \phi^*\|^2 \geq \frac{1}{\mu} \left[\|y - y^*\|_{I(W^{-1})}^2 + \|\phi - \phi^*\|^2 \right] \Rightarrow$$

$$g(y^*, \phi^*) \geq g(y^{(k)}, \phi^{(k)}) + \langle \nabla g(y^{(k)}, \phi^{(k)}), (y^* - y^{(k)}, \phi^* - \phi^{(k)}) \rangle$$

$$+ \frac{1}{\mu} \left[\|y - y^*\|_{I(W^{-1})}^2 + \|\phi - \phi^*\|^2 \right], \tag{23}$$

$$g(y^{(k)}, \phi^{(k)}) \geq g(y^*, \phi^*) + \langle \nabla g(y^*, \phi^*), (y^{(k)} - y^*, \phi^{(k)} - \phi^*) \rangle$$

$$+ \frac{1}{\mu} \left[\|y - y^*\|_{I(W^{-1})}^2 + \|\phi - \phi^*\|^2 \right]. \tag{24}$$

As $\langle \nabla g(y^*, \phi^*), (y^{(k)} - y^*, \phi^{(k)} - \phi^*) \rangle \geq 0$, (24) gives

$$g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)}) \leq -\frac{1}{\mu} \left[\|y - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi^*\|^2 \right]. \tag{25}$$

Inequality (23) establishes Claim 1) in (22). Claim 2) in (22) follows from (25). ∎

The following lemma is a direct consequence of the optimality of $y_r^{(k+1)}$ as the projection Π_{C_r}.

Lemma 22 *One has*

$$\langle \nabla_r g((y^{(k)}, \phi^{(k)})), (y_r^{(k+1)} - y_r^*, \phi_r^{(k+1)} - \phi_r^*) \rangle$$
$$\leq 2\langle y_r^{(k)} - y_r^{(k+1)}, y_r^{(k+1)} - y_r^* \rangle_{W^{-1}} + 2\langle \phi_r^{(k)} - \phi_r^{(k+1)}, \phi_r^{(k+1)} - \phi_r^* \rangle.$$

The following lemma follows from a simple manipulation of the Euclidean norm.

Lemma 23 *It holds that*

$$\|y_r^{(k+1)} - y_r^{(k)}\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^{(k)})^2$$
$$= \|y_r^{(k+1)} - y_r^*\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^*)^2 + \|y_r^{(k)} - y_r^*\|_{W^{-1}}^2 + (\phi_r^{(k)} - \phi_r^*)^2$$
$$+ 2\langle y_r^{(k+1)} - y_r^*, y_r^* - y_r^{(k)} \rangle_{W^{-1}} + 2\langle \phi_r^{(k+1)} - \phi_r^*, \phi_r^* - \phi_r^{(k)} \rangle$$
$$= -\|y_r^{(k+1)} - y_r^*\|_{W^{-1}}^2 - (\phi_r^{(k+1)} - \phi_r^*)^2 + \|y_r^{(k)} - y_r^*\|_{W^{-1}}^2 + (\phi_r^{(k)} - \phi_r^*)^2$$
$$+ 2\langle y_r^{(k+1)} - y_r^*, y_r^{(k+1)} - y_r^{(k)} \rangle_{W^{-1}} + 2\langle \phi_r^{(k+1)} - \phi_r^*, \phi_r^{(k+1)} - \phi_r^{(k)} \rangle.$$

Our next task is to determine how the objective function decreases in each iteration. The following expectation is with respect to uniformly sampled values of $r \in [R]$ in the k-th iteration:

$$\mathbb{E}\left[g(y^{(k+1)}, \phi^{(k+1)})\right] - g(y^{(k)}, \phi^{(k)})$$
$$= \mathbb{E}\left[\langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^{(k+1)} - y_r^{(k)}, \phi_r^{(k+1)} - \phi_r^{(k)}) \rangle + \|y_r^{(k+1)} - y_r^{(k)}\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^{(k)})^2\right]$$
$$= \mathbb{E}\left[\langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^* - y_r^{(k)}, \phi_r^* - \phi_r^{(k)}) \rangle + \langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^{(k+1)} - y_r^*, \phi_r^{(k+1)} - \phi_r^*) \rangle\right.$$
$$\left. + \|y_r^{(k+1)} - y_r^{(k)}\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^{(k)})^2\right]$$
$$\overset{1)}{\leq} \mathbb{E}\left[\langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^* - y_r^{(k)}, \phi_r^* - \phi_r^{(k)}) \rangle - \|y_r^{(k+1)} - y_r^*\|_{W^{-1}}^2 + \|y_r^* - y_r^{(k)}\|_{W^{-1}}^2\right.$$
$$\left. - (\phi_r^{(k+1)} - \phi_r^*)^2 + (\phi_r^* - \phi_r^{(k)})^2\right]$$
$$\overset{2)}{\leq} \frac{1}{R}\langle \nabla g(y^{(k)}, \phi^{(k)}), (y^* - y^{(k)}, \phi^* - \phi^{(k)}) \rangle - \mathbb{E}\left[\|y^{(k+1)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k+1)} - \phi^*\|^2\right]$$
$$+ \|y^{(k)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi^*\|^2 \tag{26}$$
$$\overset{3)}{\leq} \frac{2}{(\mu+1)R}\left[g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)})\right] + \left(1 - \frac{2}{(\mu+1)R}\right)\left[\|y^{(k)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi^*\|^2\right]$$
$$- \mathbb{E}\left[\|y^{(k+1)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k+1)} - \phi^*\|^2\right]. \tag{27}$$

Here, 1) is a consequence of Lemma 22 and Lemma 23, 2) is due to $y_{r'}^{(k+1)} = y_{r'}^{(k)}$, $\phi_{r'}^{(k+1)} = \phi_{r'}^{(k)}$ for $r' \neq r$, and 3) may be established from (22).

Equation (35) further leads to

$$
\mathbb{E}\left[g(y^{(k+1)}, \phi^{k+1}) - g(y^*, \phi^*) + d^2((y^{(k+1)}, \phi^{(k+1)}), \Xi)\right]
$$

$$
\leq \quad \mathbb{E}\left[g(y^{(k+1)}, \phi^{k+1}) - g(y^*, \phi^*) + \|y^{(k+1)} - y^*\|^2_{I(W^{-1})} + \|\phi^{(k+1)} - \phi^*\|^2_{I(W^{-1})}\right]
$$

$$
\leq \quad \left[1 - \frac{2}{(\mu + 1)R}\right] \mathbb{E}\left[g(y^{(k)}, \phi^k) - g(y^*, \phi^*) + d^2((y^{(k)}, r^{(k)}), \Xi)\right].
$$

The claimed proof follows by iterating the above derivations for all values of k.

7.3 Convergence Analysis of the AP Algorithm

7.3.1 Proof of Lemma 5

First, for $r \in [R]$, we define a diagonal matrix $A_r \in \mathbb{R}^{N \times N}$: $(A_r)_{ii} = 1$, if $i \in S_r$, and 0 otherwise. Start with a Lagrangian dual of (11), and transform it according to

$$
\min_{(y,\phi) \in \bigotimes_{r \in [R]} C_r} \min_{\Lambda : \lambda_{r,i} = 0, \forall (i,r) \notin \mathbb{S}} \max_{\alpha \in \mathbb{R}^N} \sum_{r \in [R]} \left[\|y_r - \lambda_r\|^2_{\Psi W^{-1}} + \phi_r^2\right] + \left\langle \alpha, \sum_{r \in [R]} \lambda_r - 2Wa \right\rangle
$$

$$
= \min_{(y,\phi) \in \bigotimes_{r \in [R]} C_r} \max_{\alpha \in \mathbb{R}^N} \min_{\Lambda : \lambda_{r,i} = 0, \forall (i,r) \notin \mathbb{S}} \sum_{r \in [R]} \left[\|y_r - \lambda_r\|^2_{\Psi W^{-1}} + \phi_r^2\right] + \left\langle \alpha, \sum_{r \in [R]} \lambda_r - 2Wa \right\rangle
$$

$$
\overset{1)}{=} \min_{(y,\phi) \in \bigotimes_{r \in [R]} C_r} \max_{\alpha \in \mathbb{R}^N} \sum_{r \in [R]} \left[\frac{1}{4}\|A_r \Psi^{-1} W\alpha\|^2_{\Psi W^{-1}} + \phi_r^2\right]
$$

$$
+ \left\langle \alpha, \sum_{r \in [R]} \left(y_r - \frac{1}{2}A_r \Psi^{-1} W\alpha\right) - 2Wa \right\rangle
$$

$$
\overset{2)}{=} \min_{(y,\phi) \in \bigotimes_{r \in [R]} C_r} \max_{\alpha \in \mathbb{R}^N} -\frac{1}{4}\|\alpha\|^2_W + \left\langle \alpha, \sum_{r \in [R]} y_r - 2Wa \right\rangle + \sum_{r \in [R]} \phi_r^2
$$

$$
= \min_{(y,\phi) \in \bigotimes_{r \in [R]} C_r} \|\sum_{r \in [R]} y_r - 2Wa\|^2_{W^{-1}} + \sum_{r \in [R]} \phi_r^2.
$$

Here, 1) is due to $\lambda_r = y_r - \frac{1}{2}A_r \Psi^{-1} W\alpha$ while 2) is based on the fact that $\Psi = \sum_{r \in [R]} A_r$. This establishes the claimed result.

7.3.2 Proof of Lemma 8

Suppose that $(y, \phi) \in \mathcal{C}/\Xi$. Then,

$$
[d_{\Psi W^{-1}}((y, \phi), \mathcal{Z})]^2 = \min_{\lambda_r, \forall r \in [R]} \sum_r \left[\|y_r - \lambda_r\|^2_{\Psi W^{-1}} + (\phi_r - \phi_r^*)^2\right]
$$

$$
\text{s.t.} \sum_{r \in [R]} \lambda_r = 2W(a - x^*), \lambda_{r,i} = 0, \forall r \in [R], i \notin S_r.
$$

By eliminating λ_r, we arrive at

$$
[d_{\Psi W^{-1}}((y, \phi), \mathcal{Z})]^2 = \|\sum_r y_r - 2W(a - x^*)\|^2_{W^{-1}} + \sum_r (\phi_r - \phi_r^*)^2.
$$

Based on Lemma 2, we know that there exists a $(y', \phi') \in \Xi$ such that

$$\mu(\Psi W^{-1}, W^{-1}) \left[\| \sum_r (y_r - y_r') \|_{W^{-1}}^2 + \sum_r (\phi_r - \phi_r')^2 \right] \geq \|y - y'\|_{\Psi W^{-1}}^2 + \sum_r (\phi_r - \phi_r')^2.$$

As ϕ_r^* is the unique optima, it follows that $\phi_r^* = \phi_r'$. Also, $\sum_r y_r' = 2W(a - x^*)$. Moreover, as

$$\|y - y'\|_{\Psi W^{-1}}^2 + \sum_r (\phi_r - \phi_r')^2 \geq [d_{\Psi W^{-1}}((y, \phi), \Xi)]^2,$$

according to the above definition, we have

$$\frac{[d_{\Psi W^{-1}}((y, \phi), \Xi)]^2}{[d_{\Psi W^{-1}}((y, \phi), \mathcal{Z})]^2} \leq \mu(\Psi W^{-1}, W^{-1}).$$

Next, suppose that $(y, \phi) \in \mathcal{Z}/\Xi$ and that

$$(y', \phi') = \arg \min_{(z, \psi) \in \mathcal{C}} \|y - z\|_{I(\Psi W^{-1})}^2 + \|\phi - \psi\|^2,$$

$$(y'', \phi'') = \arg \min_{(z, \psi) \in \Xi} \|y' - z\|_{I(\Psi W^{-1})}^2 + \|\phi' - \psi\|^2.$$

Again, due to the definition of the distance $d_{\Psi W^{-1}}((y, \phi), \Xi)$, we have

$$[d_{\Psi W^{-1}}((y, \phi), \Xi)]^2 \leq \|y - y''\|_{I(\Psi W^{-1})}^2 + \|\phi - \phi''\|^2. \tag{28}$$

Moreover, because of the way we chose (y', ϕ') and due to the fact that $\mathcal{C}$ is convex, we have

$$\|y - y''\|_{I(\Psi W^{-1})}^2 + \|\phi - \phi''\|^2 \leq \|y - y'\|_{I(\Psi W^{-1})}^2 + \|\phi - \phi'\|^2$$
$$+ \|y' - y''\|_{I(\Psi W^{-1})}^2 + \|\phi' - \phi''\|^2. \tag{29}$$

Using Lemma 2, we obtain

$$\|y' - y''\|_{I(\Psi W^{-1})}^2 + \|\phi' - \phi''\|^2 \leq \mu(\Psi W^{-1}, W^{-1})(\| \sum_r (y_r' - y_r'') \|_{W^{-1}}^2 + \|\phi' - \phi''\|^2), \tag{30}$$

and we also have

$$\| \sum_r (y_r' - y_r'') \|_{W^{-1}}^2 = \| \sum_r y_r' - 2W(a - x^*) \|_{W^{-1}}^2$$

$$= \| \sum_r (y_r' - y_r) \|_{W^{-1}}^2 \overset{1)}{\leq} \|(y' - y)\|_{I(\Psi W^{-1})}^2, \tag{31}$$

where 1) follows from the Cauchy-Schwarz inequality over the entries $y_{r,i}$, $i \in S_r$.

Finally, we set $\phi = \phi'' = \phi^*$ and combine (28)-(31) to obtain

$$[d_{\Psi W^{-1}}((y, \phi), \Xi)]^2 \leq (1 + \mu(\Psi W^{-1}, W^{-1}))(\|y - y'\|_{I(\Psi W^{-1})}^2 + \|\phi - \phi'\|^2)$$
$$= (1 + \mu(\Psi W^{-1}, W^{-1}))[d_{\Psi W^{-1}}((y, \phi), \mathcal{C})]^2,$$

which concludes the proof.

7.4 Proof of Corollary 9

First, we establish an upper bound on ρ.

Lemma 24 *Suppose that $D_{ii} = \sum_{r:r\in[R],i\in C_r} \max_{S\subseteq V}[F_r(S)]^2$. Then*

$$\rho^2 \leq 4 \sum_{i\in[N]} D_{ii}.$$

Proof For each r, consider $y_r \in B_r$. Sort the entries of y_r in descending order. Without loss of generality, assume that the ordering reads as $y_{r,i_1} \geq y_{r,i_2} \geq \cdots \geq y_{r,i_N}$. As $F_r([N]) = \sum_{j=1}^N y_{r,i_j} \geq 0$, we have $y_{r,i_1} \geq 0$. If $y_{r,i_N} \geq 0$, then $\|y_r\|_1 = \sum_{k=1}^N y_{r,i_k} = F_r([N]) \leq \max_{S\subseteq[N]} F_r(S)$. If $y_{r,i_N} < 0$, there exists a k' such that $y_{r,i_{k'}} \geq 0$ and $y_{r,i_{k'+1}} < 0$. Given the definition of B_r, we have

$$\sum_{k=k'+1}^N |y_{r,i_k}| \leq \sum_{k=1}^{k'} |y_{r,i_k}| \leq F_r(\{i_1, i_2, ..., i_{k'}\}) \leq \max_{S\subseteq[N]} F_r(S),$$

and thus $\|y_r\|_1 \leq 2\max_{S\subseteq[N]} F_r(S)$. Moreover, as each variable in $[N]$ is incident to at least one submodular function, we have

$$\sum_{r\in[R]} \max_{S\subseteq[N]}[F_r(S)]^2 \leq \sum_{i\in[N]} \sum_{r:i\in S_r} \max_{S\subseteq[N]}[F_r(S)]^2 \leq \sum_{i\in[N]} D_{ii}.$$

Combining all of the above results, we obtain

$$\rho^2 = \sum_{r\in[R]} \max_{y_r\in B_r} \|y_r\|_1^2 \leq 4 \sum_{r\in[R]} \max_{S\subseteq[N]}[F_r(S)]^2 \leq 4 \sum_{i\in[N]} D_{ii}.$$

■

When $W = \beta D$, we have

$$\sum_{i\in[N]} W_{ii} \sum_{j\in[N]} 1/W_{jj} \leq N^2 \max_{i,j} \frac{W_{ii}}{W_{jj}} = N^2 \max_{i,j} \frac{D_{ii}}{D_{jj}},$$

and

$$\rho^2 \sum_{j\in[N]} 1/W_{jj} \overset{1)}{\leq} 4 \sum_{i\in[N]} D_{ii} \sum_{j\in[N]} 1/W_{jj} \leq \frac{4}{\beta} N^2 \max_{i,j} \frac{D_{ii}}{D_{jj}},$$

where 1) follows from Lemma 24. According to the definition of $\mu(W^{-1}, W^{-1})$ (see (8)),

$$\mu(W^{-1}, W^{-1}) \leq N^2 \max\{1, 9\beta^{-1}\} \max_{i,j} \frac{D_{ii}}{D_{jj}}.$$

Similarly, we have

$$\mu(\Psi W^{-1}, W^{-1}) \leq N^2 \max\{1, 9\beta^{-1}\} \max_{i,j} \frac{\Psi_{jj} D_{ii}}{D_{jj}}.$$

This concludes the proof.

7.5 Convergence Analysis of the Conic MNP Algorithm

7.5.1 Preliminary Notation and Lemmas

Given an active set $S = \{q_1, q_2, ...\}$, and a collection of coefficients $\lambda = \{\lambda_1, \lambda_2, ...\}$, if $y = \sum_{q_i \in S} \lambda_i q_i$, we simply refer to (y, S, λ) as a *triple*.

Define the following functions that depend on S

$$\tilde{h}(S, \lambda) \triangleq h(\sum_{q_i \in S} \lambda_i q_i, \sum_{q_i \in S} \lambda_i),$$

$$\tilde{h}(S) \triangleq \min_{\lambda:\lambda_i \in \mathbb{R}, \forall i} \tilde{h}(S, \lambda),$$

$$\tilde{h}_+(S) \triangleq \min_{\lambda:\lambda_i \geq 0, \forall i} \tilde{h}(S, \lambda).$$

If the coefficients λ ($\lambda_i \in \mathbb{R}, \forall i$) minimize $\tilde{h}(S, \lambda)$, we call the corresponding triple (y, S, λ) a *good* triple. Given a triple (y, S, λ), we also define

$$\triangle(y, q) = -\langle y - a, q \rangle - \sum_{q_i \in S} \lambda_i,$$

$$\triangle(y) = \max_{q \in B} \triangle(y) = -\min_{q \in B}\langle y - a, q \rangle - \sum_{q_i \in S} \lambda_i,$$

and

$$\text{err}(y) = h(y, \sum_{q_i \in S} \lambda_i) - h^*.$$

The following lemma establishes the optimality of a good triple.

Lemma 25 *Given an active set S, consider the good triple (y', S, λ') and an arbitrary triple (y, S, λ). Then,*

$$\langle y' - a, y \rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda'_i, \sum_{q_i \in S} \lambda_i \rangle = \langle y' - a, y' - y \rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda'_i, \sum_{q_i \in S} (\lambda'_i - \lambda_i) \rangle = 0.$$

Proof Without loss of generality, assume that $\langle y' - a, y \rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda'_i, \sum_{q_i \in S} \lambda_i \rangle < 0$. Then, for any $\epsilon > 0$, $(y' + \epsilon y, S, \lambda' + \epsilon \lambda)$ is also a triple. For ϵ sufficiently small, we have $\tilde{h}(S, \lambda' + \epsilon \lambda) < \tilde{h}(S, \lambda')$, which contradicts the optimality of (y', S, λ'). Hence,

$$\langle y' - a, y \rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda'_i, \sum_{q_i \in S} \lambda_i \rangle = 0.$$

As $(y' - y, S, \lambda' - \lambda)$ is also a triple, repeating the above procedure we obtain the claimed equality. ∎

Lemma 26 *For any $\hat{y} \in B$,*

$$\arg\min_{\phi \geq 0} h(\phi \hat{y}, \phi) \leq \frac{\|a\|_{\tilde{W}}}{2}.$$

Moreover, $\phi^* \leq \frac{\|a\|_{\tilde{W}}}{2}$.

Proof Given $\hat{y}$, the optimal value of ϕ satisfies

$$\phi = \frac{\langle a, \hat{y} \rangle_{\tilde{W}}}{1 + \|\hat{y}\|^2_{\tilde{W}}} \leq \frac{\|a\|_{\tilde{W}}}{\|\hat{y}\|_{\tilde{W}} + \frac{1}{\|\hat{y}\|_{\tilde{W}}}} \leq \frac{\|a\|_{\tilde{W}}}{2}.$$

This establishes the claimed result. ∎

Lemma 27 *If (y, S, λ) is a good triple, then $\triangle(y) \geq \frac{err(y)}{\|a\|_{\tilde{W}}}$, where we recall that $err(y) = h(y, \sum_{q_i \in S} \lambda_i) - h^*$.*

Proof Recall that (y^*, ϕ^*) denotes the optimal solution. As $y^*/\phi^* \in B$, we have

$$\phi^* \triangle(y) \geq -\langle y - a, y^* \rangle_{\tilde{W}} - \langle \phi^*, \sum_{q_i \in S} \lambda_i \rangle$$

$$\overset{1)}{=} -\langle y - a, y^* \rangle_{\tilde{W}} - \langle \phi^*, \sum_{q_i \in S} \lambda_i \rangle + \langle y - a, y \rangle_{\tilde{W}} + (\sum_{q_i \in S} \lambda_i)^2$$

$$= -\langle y - a, y^* - a \rangle_{\tilde{W}} - \langle \phi^*, \sum_{q_i \in S} \lambda_i \rangle + \langle y - a, y - a \rangle_{\tilde{W}} + (\sum_{q_i \in S} \lambda_i)^2$$

$$\overset{2)}{\geq} \frac{1}{2} \left[\|y - a\|^2_{\tilde{W}} + (\sum_{q_i \in S} \lambda_i)^2 - \|y^* - a\|^2_{\tilde{W}} + (\phi^*)^2 \right]$$

$$= \frac{1}{2} err(y),$$

where 1) follows from Lemma 25, while 2) is a consequence of the Cauchy-Schwarz inequality. By using the bound for ϕ^* described in Lemma 26, we arrive at the desired conclusion. ∎

7.5.2 PROOF OF THEOREM 10

We only need to prove the following three lemmas which immediately give rise to Theorem 10. Lemma 28 corresponds to the first statement of Theorem 10. Combining Lemma 29 and Lemma 30, we can establish the second statement of Theorem 10. This follows as we may choose $\epsilon = \delta \|a\|_{\tilde{W}}$.

If Algorithm 2 terminates after less than $O(N\|a\|^2_{\tilde{W}} \max\{Q^2, 1\}/\epsilon)$ iterations, then the condition of Lemma 29 is satisfied and thus $err(y^{(k)}) \leq \epsilon = \delta \|a\|_{\tilde{W}}$. If Algorithm 2 does not terminate after $O(N\|a\|^2_{\tilde{W}} \max\{Q^2, 1\}/\epsilon)$ iterations, Lemma 30 guarantees $err(y^{(k)}) \leq \epsilon = \delta \|a\|_{\tilde{W}}$.

Lemma 28 *At any point before Algorithm 2 terminates, one has*

$$h(y^{(k)}, \phi^{(k)}) \geq h(y^{(k+1)}, \phi^{(k+1)});$$

moreover, if $(y^{(k)}, \phi^{(k)})$ triggers a MAJOR loop, the claimed inequality is strict.

The following lemma characterizes the pair (y, ϕ) at the point when the MNP method terminates.

Lemma 29 *In the MAJOR loop at iteration k, if $\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} \geq -\delta$, then $h(y^{(k)}, \phi^{(k)}) \leq h^* + \|a\|_{\tilde{W}} \delta$.*

Lemma 30 *If Algorithm 2 does not terminate, then for any $\epsilon > 0$, one can guarantee that after $O(N\|a\|_{\tilde{W}}^2 \max\{Q^2, 1\}/\epsilon)$ iterations, Algorithm 2 generates a pair (y, ϕ) that satisfies $err(y) \leq \epsilon$.*

The proofs of Lemma 28 and Lemma 29 are fairly straightforward, while the proof of Lemma 30 is significantly more involved and postponed to the next section.

Proof of Lemma 28. Suppose that $(y^{(k)}, \phi^{(k)})$ starts a MAJOR loop. As

$$\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} < -\delta,$$

we know that there exists some small ε such that

$$h(y^{(k)} + \varepsilon q^{(k)}, \phi^{(k)} + \varepsilon) < h(y^{(k)}, \phi^{(k)}).$$

Consider next the relationship between $(z^{(k)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)$ and $(y^{(k)}, \phi^{(k)})$. Because of Step 6, we know that

$$h(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i) = \tilde{h}(S^{(k)} \cup \{q^{(k)}\}) \leq h(y^{(k)} + \varepsilon q^{(k)}, \phi^{(k)} + \varepsilon) < h(y^{(k)}, \phi^{(k)}).$$

If $(y^{(k+1)}, \phi^{(k+1)})$ is generated in some MAJOR loop, then

$$(y^{(k+1)}, \phi^{(k+1)}) = (z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i),$$

which naturally satisfies the claimed condition. If $(y^{(k+1)}, \phi^{(k+1)})$ is generated in some MINOR loop, then $(y^{(k+1)}, \phi^{(k+1)})$ lies strictly within the segment between $(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)$ and $(y^{(k)}, \phi^{(k)})$ (because $\theta > 0$). Therefore, we also have $h(y^{(k+1)}, \phi^{(k+1)}) < h(y^{(k)}, \phi^{(k)})$. If $(y^{(k)}, \phi^{(k)})$ starts a MINOR loop, then we have

$$h(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i) = \tilde{h}(S^{(k)}) \leq h(y^{(k)}, \phi^{(k)}),$$

once again due to Step 6. As $(y^{(k+1)}, \phi^{(k+1)})$ still lies within the segment between $(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)$ and $(y^{(k)}, \phi^{(k)})$, we have $h(y^{(k+1)}, \phi^{(k+1)}) \leq h(y^{(k)}, \phi^{(k)})$.

Proof of Lemma 29. Lemma 29 is a corollary of Lemma 27. To see why this is the case, observe that in a MAJOR loop, $(y^{(k)}, S^{(k)}, \lambda^{(k)})$ is always a good triple. Since

$$\triangle(y^{(k)}) = -(\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)}) \leq \delta,$$

we have $err(y) \leq \delta\|a\|_{\tilde{W}}$.

7.5.3 PROOF OF LEMMA 30

The outline of the proof is similar to that of the standard case described in (Chakrabarty et al., 2014), and some results therein can be directly reused. The key step is to show that in every MAJOR loop k with no more than one MINOR loop, the objective achieved by $y^{(k)}$ decreases sufficiently, as precisely described in Theorem 31.

Theorem 31 *For a MAJOR loop with no more than one MINOR loop, if the starting point is y, the starting point y' of the next MAJOR loop satisfies*

$$err(y') \leq err(y) \left(1 - \frac{err(y)}{\|a\|_{\tilde{W}}(Q^2 + 1)} \right).$$

Based on this theorem, it is easy to establish the result in Lemma 30 by using the next lemma and the same approach as described in (Chakrabarty et al., 2014).

Lemma 32 (Lemma 1 (Chakrabarty et al., 2014)) *In any consecutive $3N + 1$ iteratons, there exists at least one MAJOR loop with not more than one MINOR loop.*

We now focus on the proof of Theorem 31. The next geometric lemma is the counterpart of Lemma 2 (Chakrabarty et al., 2014) for the conic case.

Lemma 33 *Given an active set S, consider a good triple (y', S, λ') and an arbitrary triple (y, S, λ). Then, for any $q \in lin(S)$ such that $\triangle(y, q) > 0$, we have*

$$\|y - a\|_{\tilde{W}}^2 + (\sum_{q_i \in S} \lambda_i)^2 - \|y' - a\|_{\tilde{W}}^2 - (\sum_{q_i \in S} \lambda_i')^2 \geq \frac{\triangle^2(y, q)}{\|q\|_{\tilde{W}}^2 + 1}.$$

Proof First, we have

$$\|y - a\|_{\tilde{W}}^2 + (\sum_{q_i \in S} \lambda_i)^2 - \|y' - a\|_{\tilde{W}}^2 - (\sum_{q_i \in S} \lambda_i')^2$$

$$= \|y - y'\|_{\tilde{W}}^2 + [\sum_{q_i \in S}(\lambda_i - \lambda_i')]^2 + 2\langle y - y', y' - a\rangle_{\tilde{W}} + 2\langle \sum_{q_i \in S}(\lambda_i - \lambda_i'), \sum_{q_i \in S} \lambda_i'\rangle$$

$$\overset{1)}{=} \|y - y'\|_{\tilde{W}}^2 + [\sum_{q_i \in S}(\lambda_i - \lambda_i')]^2,$$

where 1) follows from Lemma 25. Next, for any $\phi \geq 0$,

$$\|y - y'\|_{\tilde{W}}^2 + [\sum_{q_i \in S}(\lambda_i - \lambda_i')]^2$$

$$\overset{1)}{\geq} \frac{\left[\langle y - y', y - \phi q\rangle_{\tilde{W}} + \langle \sum_{q_i \in S}(\lambda_i - \lambda_i'), \sum_{q_i \in S} \lambda_i - \phi\rangle \right]^2}{\|y - \phi q\|_{\tilde{W}}^2 + (\sum_{q_i \in S} \lambda_i - \phi)^2}$$

$$= \frac{\left[\langle y - a, y - \phi q\rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda_i, \sum_{q_i \in S} \lambda_i - \phi\rangle - \langle y' - a, y - \phi q\rangle_{\tilde{W}} - \langle \sum_{q_i \in S} \lambda_i', \sum_{q_i \in S} \lambda_i - \phi\rangle \right]^2}{\|y - rq\|_{\tilde{W}}^2 + (\sum_{q_i \in S} \lambda_i - \phi)^2}$$

$$\overset{2)}{=} \frac{\left[\langle y - a, y - \phi q\rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda_i, \sum_{q_i \in S} \lambda_i - \phi\rangle \right]^2}{\|y - \phi q\|_{\tilde{W}}^2 + (\sum_{q_i \in S} \lambda_i - \phi)^2}, \tag{32}$$

where 1) follows from the Cauchy-Schwarz inequality and 2) is due to Lemma 25. Since $\triangle(y, q) > 0$, letting $\phi \to \infty$ reduces equation (32) to $\frac{\triangle^2(y,q)}{\|q\|_{\tilde{W}}^2 + 1}$. ∎

Next, using Lemma 33, we may characterize the decrease of the objective function for one MAJOR loop with no MINOR loop. As $(y^{(k+1)}, S^{(k+1)}, \lambda^{(k+1)})$ is a good triple and $y^{(k)}$ also lies in $\text{lin}(S)$, we have the following result.

Lemma 34 *Consider some MAJOR loop k without MINOR loops. Then*

$$err(y^{(k)}) - err(y^{(k+1)}) \geq \frac{\triangle^2(y^{(k)}, q^{(k)})}{Q^2 + 1} = \frac{\triangle^2(y^{(k)})}{Q^2 + 1}.$$

Next, we characterize the decrease of the objective function for one MAJOR loop with one single MINOR loop.

Lemma 35 *Consider some MAJOR loop k with only one MINOR loop. Then*

$$err(y^{(k)}) - err(y^{(k+2)}) \geq \frac{\triangle^2(y^{(k)})}{Q^2 + 1}.$$

Proof Suppose that the active sets associated with $y^{(k)}, y^{(k+1)}, y^{(k+2)}$ are $S^{(k)}, S^{(k+1)}, S^{(k+2)}$, respectively. We know that within the MINOR loop, $(z^{(k)}, S^{(k)} \cup \{q^{(k)}\}, \alpha)$ is a good triple and $y^{(k+1)} = \theta y^{(k)} + (1 - \theta)z^{(k)}$, for some $\theta \in [0, 1]$. Let

$$A = \|y^{(k)} - a\|_{\tilde{W}}^2 + \left(\sum_{q_i \in S^{(k)}} \lambda_i^{(k)} \right)^2 - \|z^{(k)} - a\|_{\tilde{W}}^2 - \left(\sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i \right)^2. \tag{33}$$

From Lemma 33, we have

$$A \geq \frac{\triangle^2(y^{(k)}, q^{(k)})}{\|q^{(k)}\|_{\tilde{W}}^2 + 1} \geq \frac{\triangle^2(y^{(k)})}{Q^2 + 1}. \tag{34}$$

Note that both $S^{(k)}$ and $S^{(k+1)}$ are subsets of $S^{(k)} \cup \{q^{(k)}\}$. As $(z^{(k)}, S^{(k)} \cup \{q^{(k)}\}, \alpha)$ is a good triple, using Lemma 25, we obtain

$$\langle z^{(k)} - a, z^{(k)} - y^{(k)} \rangle_{\tilde{W}} + \langle \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} (\alpha_i - \lambda_i^{(k)}) \rangle = 0.$$

Furthermore, as $y^{(k+1)} = \theta y^{(k)} + (1-\theta)z^{(k)} = z^{(k)} - \theta(z^{(k)} - y^{(k)})$ and $\lambda^{(k+1)} = \alpha - \theta(\alpha - \lambda^{(k)})$, we have

$$\|y^{(k)} - a\|_{\tilde{W}}^2 + \left(\sum_{i \in S^{(k)}} \lambda_i^{(k)} \right)^2 - \|y^{(k+1)} - a\|_{\tilde{W}}^2 - \left(\sum_{i \in S^{(k+1)}} \lambda_i^{(k+1)} \right)^2 = (1 - \theta^2)A.$$

Moreover, we have

$$\triangle(y^{(k+1)}, q^{(k)}) = \theta \triangle(y^{(k)}, q^{(k)}) + (1 - \theta)\triangle(z^{(k)}, q^{(k)}) = \theta \triangle(y^{(k)}, q^{(k)}) = \theta \triangle(y^{(k)}), \tag{35}$$

which holds because $\triangle(y, q)$ is linear in y and Lemma 25 implies $\triangle(z^{(k)}, q^{(k)}) = 0$.

Since according to Lemma 28, $h(y^{(k)}, \phi^{(k)}) > h(y^{(k+1)}, \phi^{(k+1)})$, Lemma 4 in (Chakrabarty et al., 2014) also holds in our case, and thus $q^{(k)} \in S^{(k+1)}$. To obtain $y^{(k+2)}$ and $S^{(k+2)}$, one needs to remove active points with a zero coefficients from $S^{(k+1)}$, so that $y^{(k+2)}$ once again belongs to a good triple with corresponding $S^{(k+1)}$. Based on 33 and equation (35), we have the following result.

$$\|y^{(k+1)} - a\|_{\tilde{W}}^2 + (\sum_{q_i \in S^{(k+1)}} \lambda_i^{(k+1)})^2 - \|y^{(k+2)} - a\|_{\tilde{W}}^2 - (\sum_{q_i \in S^{(k+2)}} \lambda_i^{(k+2)})^2 \tag{36}$$

$$\geq \frac{\triangle^2(y^{(k+1)}, q^{(k)})}{Q^2 + 1} = \frac{\theta^2 \triangle^2(y^{(k)})}{Q^2 + 1}. \tag{37}$$

Consequently, combining equations (33), (34) and (36), we arrive at

$$\text{err}(y^{(k)}) - \text{err}(y^{(k+2)}) \geq \frac{\triangle^2(y^{(k)})}{Q^2 + 1}.$$

∎

And, combining Lemma 34, Lemma 35 and Lemma 27 establishes Theorem 31.

7.6 Convergence Analysis of the Conic Frank-Wolfe Algorithm

7.6.1 Proof of Theorem 11

Using the same strategy as in the proof of Lemma 26, we may prove the following lemma. It hinges on the optimality assumption for $\gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)}$ in Step 3 of Algorithm 3.

Lemma 36 *In Algorithm 3, for all k, $\phi^{(k+1)} \leq \frac{\|a\|_{\tilde{W}}}{2}$.*

Now, we prove Theorem 11. We write $y = \phi \hat{y}$, where $\hat{y} \in B$, so that

$$h(y, \phi) = h(\phi \hat{y}, \phi) \geq h(y^{(k)}, \phi^{(k)}) + 2\langle y^{(k)} - a, \phi \hat{y} - y^{(k)} \rangle_{\tilde{W}} + \phi^2 - (\phi^{(k)})^2.$$

For both sides, minimize $(\hat{y}, \phi)$ over $B \times [0, \frac{\|a\|_{\tilde{W}}}{2}]$, which contains (y^*, ϕ^*). Since $q^{(k)} = \arg \min_{q \in B}(y^{(k)} - a, q)_{\tilde{W}}$, we know that the optimal solutions satisfy $\hat{y} = q^{(k)}$ and $\phi = \tilde{\phi}^{(k)} = \min\{\max\{0, -\langle y^{(k)} - a, q^{(k)} \rangle\}, \frac{\|a\|_{\tilde{W}}}{2}\}$ of the RHS

$$h^* = h(y^*, \phi^*) \geq h(y^{(k)}, \phi^{(k)}) + 2\langle y^{(k)} - a, \tilde{\phi}^{(k)} q^{(k)} - y^{(k)} \rangle_{\tilde{W}} + (\tilde{\phi}^{(k)})^2 - (\phi^{(k)})^2. \tag{38}$$

Moreover, because of the optimality of $(\gamma_1^{(k)} \times \gamma_2^{(k)}) \in \mathbb{R}_{\geq 0}^2$, for arbitrary $\gamma \in [0, 1]$ we have

$$h(\gamma_1^{(k)} y^{(k)} + \gamma_2^{(k)} q^{(k)}, \gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)})$$
$$\leq h((1 - \gamma)y^{(k)} + \gamma \tilde{\phi}^{(k)} q^{(k)}, (1 - \gamma)\phi^{(k)} + \gamma \tilde{\phi}^{(k)})$$
$$= h(y^{(k)}, \phi^{(k)}) + 2\gamma \langle y^{(k)} - a, \tilde{\phi}^{(k)} q^{(k)} - y^{(k)} \rangle_{\tilde{W}}$$
$$+ \gamma[(\tilde{\phi}^{(k)})^2 - (\phi^{(k)})^2] + \gamma^2 \|\tilde{\phi}^{(k)} q^{(k)} - y^{(k)}\|_{\tilde{W}}^2 + (\gamma^2 - \gamma)(\tilde{\phi}^{(k)} - \phi^{(k)})^2$$
$$\overset{1)}{\leq} h(y^{(k)}, \phi^{(k)}) + \gamma(h^* - h(y^{(k)}, \phi^{(k)})) + \gamma^2 \|\tilde{\phi}^{(k)} q^{(k)} - y^{(k)}\|_{\tilde{W}}^2$$
$$\overset{2)}{\leq} h(y^{(k)}, \phi^{(k)}) + \gamma(h^* - h(y^{(k)}, \phi^{(k)})) + \gamma^2 \|a\|_{\tilde{W}}^2 Q^2,$$

where 1) follows from (38) and $\gamma^2 - \gamma \leq 0$, and 2) follows from

$$\|\tilde{\phi}^{(k)} q^{(k)} - y^{(k)}\|_{\tilde{W}}^2 \leq 4 \frac{\|a\|_{\tilde{W}}^2}{4} \max_{q \in B} \|q\|_{\tilde{W}}^2 = \|a\|_{\tilde{W}}^2 Q^2.$$

The claimed result now follows by induction. First, let $\hat{y}^* = y^*/\phi^*$, where $\phi^* = \frac{\langle \hat{y}^*, a \rangle_{\tilde{W}}}{1 + \|\hat{y}^*\|_{\tilde{W}}^2}$. Then,

$$h(y^{(0)}, \phi^{(0)}) - h^* \leq 2\langle y^*, a \rangle - (y^*)^2 - (\phi^*)^2 = \frac{\langle \hat{y}^*, a \rangle_{\tilde{W}}^2}{1 + \|\hat{y}^*\|_{\tilde{W}}^2} \leq \|a\|_{\tilde{W}}^2 Q^2.$$

Suppose that $h(y^{(k)}, r^{(k)}) - h^* \leq \frac{2\|a\|_{\tilde{W}}^2 Q^2}{k+2}$. In this case, for all $\gamma \in [0, 1]$, we have

$$h(y^{(t+1)}, \phi^{(t+1)}) - h^* \leq (1 - \gamma)[h(y^{(k)}, \phi^{(k)}) - h^*] + \gamma^2 \|a\|_{\tilde{W}}^2 Q^2.$$

By choosing $\gamma = \frac{1}{k+2}$, we obtain $h(y^{(k+1)}, \phi^{(k+1)}) - h^* \leq \frac{2\|a\|_{\tilde{W}}^2 Q^2}{k+3}$, which concludes the proof.

7.7 Proofs for the Partitioning Properties of PageRank

7.7.1 PROOF OF LEMMA 14

Based on the definitions of $\mathcal{S}_j^p$ and $j \in V_p$, we have $\langle \nabla f_r(x), 1_{\mathcal{S}_j^p} \rangle = F_r(\mathcal{S}_j^p)$. Consequently,

$$2p(\mathcal{S}_j^p) - \sum_{r \in [R]} w_r f_r(x) \langle \nabla f_r(x), 1_{\mathcal{S}_j^p} \rangle$$

$$= 2p(\mathcal{S}_j^p) - \sum_{r \in [R]} w_r f_r(x) F_r(\mathcal{S}_j^p)$$

$$= 2p(\mathcal{S}_j^p) - \sum_{r \in [R]} w_r F_r(S_j^p) \max_{(i,j) \in S_r^\uparrow \times S_r^\downarrow} (x_i - x_j)$$

$$= \left(I_p(\text{vol}(\mathcal{S}_j^p)) - \sum_{r \in [R]} w_r F_r(\mathcal{S}_j^p) \max_{i \in S_r^\uparrow} x_i \right) + \left(I_p(\text{vol}(\mathcal{S}_j^p)) + \sum_{r \in [R]} w_r F_r(\mathcal{S}_j^p) \min_{i \in S_r^\downarrow} x_i \right)$$

$$\leq I_p \left(\text{vol}(\mathcal{S}_j^p) - \sum_{r \in [R]} w_r F_r(\mathcal{S}_j^p) \right) + I_p \left(\text{vol}(\mathcal{S}_j^p) + \sum_{r \in [R]} w_r F_r(\mathcal{S}_j^p) \right)$$

$$= I_p \left(\text{vol}(\mathcal{S}_j^p) - \text{vol}(\partial \mathcal{S}_j^p) \right) + I_p \left(\text{vol}(\mathcal{S}_j^p) + \text{vol}(\partial \mathcal{S}_j^p). \right) \tag{39}$$

Using equation (18), we have

$$p(\mathcal{S}_j^p) = \frac{\alpha}{2 - \alpha} p_0(\mathcal{S}_j^p) + \frac{2 - 2\alpha}{2 - \alpha} \left\{ \frac{1}{2} \left[M(p) - p \right] (\mathcal{S}_j^p) + p(\mathcal{S}_j^p) \right\}$$

$$\overset{1)}{=} \frac{\alpha}{2 - \alpha} p_0(\mathcal{S}_j^p) + \frac{2 - 2\alpha}{2 - \alpha} \left[-\frac{1}{2} \sum_{r \in [R]} f_r(x) \langle \nabla f_r(x), 1_{\mathcal{S}_j^p} \rangle + p(\mathcal{S}_j^p) \right]$$

$$\overset{2)}{\leq} \frac{\alpha}{2 - \alpha} p_0(\mathcal{S}_j^p) + \frac{1 - \alpha}{2 - \alpha} \left[I_p \left(\text{vol}(\mathcal{S}_j^p) - \text{vol}(\partial \mathcal{S}_j^p) \right) + I_p \left(\text{vol}(\mathcal{S}_j^p) + \text{vol}(\partial \mathcal{S}_j^p) \right) \right],$$

where 1) is due to Lemma 12 and 2) is due to equation (39). This proves the first inequality. By using the concavity of $I_p(\cdot)$, we also have

$$I_p(\mathrm{vol}(\mathcal{S}_j^p)) \le p_0(\mathcal{S}_j^p) \le I_{p_0}(\mathrm{vol}(\mathcal{S}_j^p)).$$

Moreover, as I_p is piecewise linear, the proof follows.

7.7.2 PROOF OF THEOREM 16

This result can be proved in a similar way as the corresponding case for graphs (Andersen et al., 2006), by using induction.

Define $\bar{k} = \min\{k, m - k\}$, $d_{\min} = \min_{i:(p_0)_i > 0} d_i$ and

$$I^{(t)}(k) = \frac{k}{m} + \frac{\alpha}{2 - \alpha} t + \sqrt{\frac{\bar{k}}{d_{\min}}} \left(1 - \frac{\Phi_p^2}{8} \right)^t.$$

When $t = 0$, $I_p(k) \le I_{p_0}(k)$ holds due to Lemma 14. As $I^{(0)}(m) = I^{(0)}(d_{\min}) = 1$, for $k \in [d_{\min}, m]$, we have

$$I_{p_0}(k) \le 1 \le I^{(0)}(k).$$

Since for $k \in [0, d_{\min}]$, $I_{p_0}(k)$ is linear, we also have $I_{p_0}(0) = 0 \le I^{(t)}(0)$. Hence, for $k \in [0, d_{\min}]$, it also holds that $I_{p_0}(k) \le I^{(0)}(k)$.

Next, suppose that for step t, $I_p(k) \le I^{(t)}(k)$ holds. We then consider the case $t + 1$. For $k = \mathrm{vol}(\mathcal{S}_j^p)$, Lemma 14 indicates that

$$I_p(k) \le \frac{\alpha}{2 - \alpha} I_{p_0}(k) + \frac{1 - \alpha}{2 - \alpha} [I_p(k - \bar{k}\Phi(\mathcal{S}_j^p)) + I_p(k + \bar{k}\Phi(\mathcal{S}_j^p))]$$

$$\overset{1)}{\le} \frac{\alpha}{2 - \alpha} I_{p_0}(k) + \frac{1 - \alpha}{2 - \alpha} \left[\frac{2k}{m} + \frac{2\alpha}{2 - \alpha} t + \left(\sqrt{\frac{k - \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} + \sqrt{\frac{k + \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} \right) \left(1 - \frac{\Phi_p^2}{8} \right)^t \right]$$

$$\overset{2)}{\le} \frac{k}{m} + \frac{\alpha}{2 - \alpha}(t + 1) + \frac{1 - \alpha}{2 - \alpha} \left(\sqrt{\frac{k - \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} + \sqrt{\frac{k + \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} \right) \left(1 - \frac{\Phi_p^2}{8} \right)^t$$

$$\overset{3)}{\le} \frac{k}{m} + \frac{\alpha}{2 - \alpha}(t + 1) + \frac{1 - \alpha}{2 - \alpha} \left(\sqrt{\frac{\bar{k} - \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} + \sqrt{\frac{\bar{k} + \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} \right) \left(1 - \frac{\Phi_p^2}{8} \right)^t$$

$$\overset{4)}{\le} \frac{k}{m} + \frac{\alpha}{2 - \alpha}(t + 1) + \frac{2 - 2\alpha}{2 - \alpha} \sqrt{\frac{\bar{k}}{d_{\min}}} \left(1 - \frac{\Phi_p^2}{8} \right)^{t+1} \le I^{(t+1)}(k),$$

where 1) follows from Lemma 14; 2) is due to $I_{p_0}(k) \le 1$; 3) can be verified by considering two cases separately, namely $k \le \frac{m}{2}$ and $k \ge \frac{m}{2}$; and 4) follows from $\Phi(\mathcal{S}_j^p) \ge \Phi_p$ and the Taylor-series expansion of $\sqrt{x \pm \phi x}$.

At this point, we have shown that $I_p(k) \le I^{(t)}(k)$ for all $k = \mathrm{vol}(\mathcal{S}_j^p)$. Since $\{\mathrm{vol}(\mathcal{S}_j^p)\}_{j \in V_p}$ covers all break points of I_p and since $I^{(t)}$ is concave, we have that for all $k \in [0, m]$, it holds that $I_p(k) \le I^{(t)}(k)$. This concludes the proof.

7.7.3 PROOF OF THEOREM 17

First, we prove that under the assumptions for the vertex-sampling probability P in the statement of the theorem, $pr(\alpha, 1_i)(S)$ can be lower bounded as follows.

Lemma 37 *If a vertex $v \in S$ is sampled according to a distribution P such that*

$$\mathbb{E}_{i \sim P}[pr(\alpha, 1_i)(\bar{S})] \leq c\, pr(\alpha, \pi_S)(\bar{S}),$$

where c is a constant, then with probability at least $\frac{1}{2}$, one has

$$pr(\alpha, 1_i)(S) \geq 1 - \frac{c\,\Phi(S)}{4\alpha}.$$

Proof Let $p = pr(\alpha, \pi_S)$. Then,

$$
\begin{aligned}
\alpha p(\bar{S}) &\overset{1)}{=} \alpha \pi_S(\bar{S}) + (1-\alpha)\left[M(p) - p\right](\bar{S}) \\
&\overset{2)}{=} -\sum_{r \in [R]} w_r f_r(x) \langle \nabla f_r(x), 1_{\bar{S}} \rangle \\
&\overset{3)}{\leq} \sum_{r \in [R]} w_r f_r(x) F(S) \leq \sum_{r \in [R]} w_r F(S) \max_{i \in S_r^\uparrow} x_i \\
&\leq I_p\Big(\sum_{r \in [R]} w_r F(S)\Big) = I_p(\text{vol}(\partial S)) \\
&\overset{4)}{\leq} I_{\pi_S}(\text{vol}(\partial S)) = \Phi(S),
\end{aligned}
$$

where 1) is a consequence of Equation (18); 2) follows from Lemma 12; 3) holds because for any $x \in \mathbb{R}^N$, $\langle \nabla f_r(x), 1_{\bar{S}} \rangle \geq -F(S)$; and 4) is a consequence of Lemma 14. Hence,

$$\mathbb{E}_{v \sim P}[pr(\alpha, 1_i)(\bar{S})] \leq \frac{c}{8} pr(\alpha, \pi_S)(\bar{S}) \leq \frac{c}{8} \frac{\Phi(S)}{\alpha}.$$

Moreover, sampling according to $v \sim P$ and using Markov's inequality, we have

$$\mathbb{P}\left[pr(\alpha, 1_i)(\bar{S}) \geq \frac{c}{4} \frac{\Phi(S)}{\alpha}\right] \leq \frac{\mathbb{E}_{v \sim P}[pr(\alpha, 1_i)(\bar{S})]}{\frac{c}{4} \frac{\Phi(S)}{\alpha}} \leq \frac{1}{2},$$

which concludes the proof. ∎

As $\Phi(S) \leq \frac{\alpha}{c}$, we have $\mathbb{P}\left[pr(\alpha, 1_i)(S) \geq \frac{3}{4}\right] \geq \frac{1}{2}$. By combining this lower bound on $pr(\alpha, 1_i)(S)$ with the upper bound of Theorem 16, we have

$$\frac{3}{4} \leq pr(\alpha, 1_i)(S) \leq I_{pr(\alpha, 1_i)}(\text{vol}(S)) \leq \frac{1}{2} + \frac{\alpha}{2-\alpha} t + \sqrt{\frac{\text{vol}(S)}{d_i}} \left(1 - \frac{\Phi_{pr(\alpha, 1_i)}^2}{8}\right)^t.$$

Next, we choose $t = \lceil \frac{8}{\Phi_{pr(\alpha, 1_i)}^2} \ln 8 \sqrt{\frac{\text{vol}(S)}{d_i}} \rceil$. Then, the above inequality may be transformed into

$$\frac{3}{4} \leq \frac{1}{2} + \frac{\alpha}{2-\alpha} \lceil \frac{8}{\Phi_{pr(\alpha, 1_i)}^2} \ln 8 \sqrt{\frac{\text{vol}(S)}{d_i}} \rceil + \frac{1}{8} \leq \frac{5}{8} + \frac{\alpha}{2-\alpha} \frac{8}{\Phi_{pr(\alpha, 1_i)}^2} \ln 10 \sqrt{\frac{\text{vol}(S)}{d_i}}.$$

Therefore,

$$\Phi_{pr(\alpha,1_i)} \leq \sqrt{32\alpha \ln \frac{100\text{vol}(S)}{d_i}}.$$

7.8 Analysis of the Parameter Choices for Experimental Verification

Let $x' = W^{-1/2}x$ and $a' = W^{-1/2}a$. We can transform the objective (21) into a standard QDSFM problem,

$$\beta\|x' - a'\|_W^2 + \sum_{r \in [R]} \max_{i,j \in S_r} (x_i' - x_j')^2.$$

From Theorem 4, to achieve an ϵ-optimal solution, one requires $O(R\mu(\beta^{-1}W^{-1}, \beta^{-1}W^{-1}) \log \frac{1}{\epsilon})$ iterations in the RCD algorithm (Algorithm 1). According to the particular settings for the experiment (undirected unweighted hypergraphs), we have

$$\rho^2 = \max_{y_r \in B_r, \forall r} \sum_{r \in [R]} \|y_r\|_1^2 = \max_{y_r \in B_r, \forall r} \sum_{r \in [R]} 2 = 2R. \tag{40}$$

From the definition of μ (8), we may rewrite $\mu(\beta^{-1}W^{-1}, \beta^{-1}W^{-1})$ as

$$\mu(\beta^{-1}W^{-1}, \beta^{-1}W^{-1}) \overset{1)}{=} \max \left\{ \frac{N^2}{2} \left(\max_{i,j \in [N]} \frac{W_{ii}}{W_{jj}} + 1 \right), \frac{9}{4}\rho^2 N\beta^{-1} \max_{j \in [N]} \frac{1}{W_{jj}} \right\}$$

$$\overset{2)}{=} \max \left\{ \frac{N^2}{2} \left(\max_{i,j \in [N]} \frac{W_{ii}}{W_{jj}} + 1 \right), \frac{9}{2}\beta^{-1} NR \max_{j \in [N]} \frac{1}{W_{jj}} \right\}$$

$$\overset{3)}{=} \max \left\{ \frac{N^2}{2} \left(\max_{i,j \in [N]} \frac{W_{ii}}{W_{jj}} + 1 \right), \frac{9}{2}\beta^{-1} N^2 \max_{i,j \in [N]} \frac{W_{ii}}{W_{jj}} \right\}, \tag{41}$$

where 1) holds because half of the values of W_{ii} are set to 1 and the other half to a value in $\{1, 0.1, 0.01, 0.001\}$, 2) follows from (40) and 3) is due to the particular setting $N = R$ and $\max_{i \in [N]} W_{ii} = 1$. Equation (41) may consequently be rewritten as

$$O(N^2 \max(1, 9/(2\beta)) \max_{i,j \in [N]} W_{ii}/W_{jj}),$$

which establishes the claimed statement.

8. Acknowledgement

References

Reid Andersen, Fan Chung, and Kevin Lang. Local graph partitioning using pagerank vectors. In *Proceedings of the 47th Annual Symposium on Foundations of Computer Science*, pages 475–486. IEEE Computer Society, 2006.

Reid Andersen, Fan Chung, and Kevin Lang. Local partitioning for directed graphs using pagerank. *Internet Mathematics*, 5(1-2):3–22, 2008.

Francis Bach. Learning with submodular functions: A convex optimization perspective. *Foundations and Trends® in Machine Learning*, 6(2-3):145–373, 2013.

Austin R Benson, David F Gleich, and Jure Leskovec. Higher-order organization of complex networks. *Science*, 353(6295):163–166, 2016.

Béla Bollobás. *Modern graph theory*, volume 184. Springer Science & Business Media, 2013.

Deeparnab Chakrabarty, Prateek Jain, and Pravesh Kothari. Provable submodular minimization using Wolfe's algorithm. In *Advances in Neural Information Processing Systems*, pages 802–809, 2014.

Antonin Chambolle and Jérôme Darbon. On total variation minimization and surface evolution using parametric maximum flows. *International Journal of Computer Vision*, 84(3):288, 2009.

Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40 (1):120–145, 2011.

T-H Hubert Chan, Zhihao Gavin Tang, Xiaowei Wu, and Chenzi Zhang. Diffusion operator and spectral analysis for directed hypergraph laplacian. *arXiv preprint arXiv:1711.01560*, 2017.

T-H Hubert Chan, Anand Louis, Zhihao Gavin Tang, and Chenzi Zhang. Spectral properties of hypergraph laplacian and approximation algorithms. *Journal of the ACM (JACM)*, 65 (3):15, 2018.

Josip Djolonga and Andreas Krause. Scalable variational inference in log-supermodular models. In *ICML*, pages 1804–1813, 2015.

Jack Edmonds. Submodular functions, matroids, and certain polyhedra. In *Combinatorial Optimization?Eureka, You Shrink!*, pages 11–26. Springer, 2003.

Ivar Ekeland and Roger Temam. *Convex Analysis and Variational Problems*, volume 28. Siam, 1999.

Alina Ene and Huy Nguyen. Random coordinate descent methods for minimizing decomposable submodular functions. In *Proceedings of the International Conference on Machine Learning*, pages 787–795, 2015.

Alina Ene, Huy Nguyen, and László A Végh. Decomposable submodular function minimization: discrete and continuous. In *Advances in Neural Information Processing Systems*, pages 2874–2884, 2017.

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics*, 3(1-2):95–110, 1956.

Satoru Fujishige. *Submodular functions and optimization*, volume 58. Elsevier, 2005.

Satoru Fujishige and Shigueo Isotani. A submodular function minimization algorithm based on the minimum-norm base. *Pacific Journal of Optimization*, 7(1):3–17, 2011.

Alexander Gammerman, Volodya Vovk, and Vladimir Vapnik. Learning by transduction. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 148–155. Morgan Kaufmann Publishers Inc., 1998.

David F Gleich, Lek-Heng Lim, and Yongyang Yu. Multilinear pagerank. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1507–1541, 2015.

Zaid Harchaoui, Anatoli Juditsky, and Arkadi Nemirovski. Conditional gradient algorithms for norm-regularized smooth convex optimization. *Mathematical Programming*, 152(1-2): 75–112, 2015.

Matthias Hein, Simon Setzer, Leonardo Jost, and Syama Sundar Rangapuram. The total variation on hypergraphs-learning on hypergraphs revisited. In *Advances in Neural Information Processing Systems*, pages 2427–2435, 2013.

Stefanie Jegelka, Francis Bach, and Suvrit Sra. Reflection methods for user-friendly submodular optimization. In *Advances in Neural Information Processing Systems*, pages 1313–1321, 2013.

Thorsten Joachims. Transductive learning via spectral graph partitioning. In *Proceedings of the 20th International Conference on Machine Learning*, pages 290–297, 2003.

Rie Johnson and Tong Zhang. On the effectiveness of laplacian normalization for graph semi-supervised learning. *Journal of Machine Learning Research*, 8(Jul):1489–1517, 2007.

Sanjiv Kumar and Martial Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1150–1157. IEEE, 2003.

Pan Li and Olgica Milenkovic. Inhomogeneous hypergraph clustering with applications. In *Advances in Neural Information Processing Systems*, pages 2305–2315, 2017.

Pan Li and Olgica Milenkovic. Revisiting decomposable submodular function minimization with incidence relations. In *Advances in Neural Information Processing Systems*, 2018a.

Pan Li and Olgica Milenkovic. Submodular hypergraphs: p-laplacians, cheeger inequalities and spectral clustering. In *Proceedings of the International Conference on Machine Learning*, 2018b.

Pan Li, Hoang Dau, Gregory Puleo, and Olgica Milenkovic. Motif clustering and overlapping clustering for social network analysis. In *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2017.

Francesco Locatello, Michael Tschannen, Gunnar Rätsch, and Martin Jaggi. Greedy algorithms for cone constrained optimization with convergence guarantees. In *Advances in Neural Information Processing Systems*, pages 773–784, 2017.

László Lovász. Submodular functions and convexity. In *Mathematical Programming The State of the Art*, pages 235–257. Springer, 1983.

László Lovasz and Miklós Simonovits. The mixing rate of markov chains, an isoperimetric inequality, and computing the volume. In *Proceedings of the 31st Annual Symposium on Foundations of Computer Science*, page 1. IEEE Computer Society, 1990.

Robert Nishihara, Stefanie Jegelka, and Michael I Jordan. On the convergence rate of decomposable submodular function minimization. In *Advances in Neural Information Processing Systems*, pages 640–648, 2014.

Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

N. Z. Shor, Krzysztof C. Kiwiel, and Andrzej Ruszcayǹski. *Minimization Methods for Non-differentiable Functions*. Springer-Verlag, Berlin, Heidelberg, 1985. ISBN 0-387-12763-1.

Peter Stobbe and Andreas Krause. Efficient minimization of decomposable submodular functions. In *Advances in Neural Information Processing Systems*, pages 2208–2216, 2010.

Philip Wolfe. Convergence theory in nonlinear programming. *Integer and nonlinear programming*, pages 1–36, 1970.

Hao Yin, Austin R Benson, Jure Leskovec, and David F Gleich. Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 555–564. ACM, 2017.

Yuichi Yoshida. Cheeger inequalities for submodular transformations. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2582–2601. Society for Industrial and Applied Mathematics, 2019.

Chenzi Zhang, Shuguang Hu, Zhihao Gavin Tang, and TH Hubert Chan. Re-revisiting learning on hypergraphs: confidence interval and subgradient method. In *Proceedings of the International Conference on Machine Learning*, pages 4026–4034, 2017.

Dawei Zhou, Si Zhang, Mehmet Yigit Yildirim, Scott Alcorn, Hanghang Tong, Hasan Davulcu, and Jingrui He. A local algorithm for structure-preserving graph cut. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 655–664. ACM, 2017.

Denny Zhou, Olivier Bousquet, Thomas N Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems*, pages 321–328, 2004.

Denny Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning with hypergraphs: Clustering, classification, and embedding. In *Advances in Neural Information Processing Systems*, pages 1601–1608, 2007.

Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning*, pages 912–919, 2003a.

Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *ICML 2003 workshop on the continuum from labeled to unlabeled data in machine learning and data mining*, volume 3, 2003b.