LARGE-SCALE REGULARIZED SUMCOR GCCA VIA PENALTY-DUAL DECOMPOSITION

Charilaos I. Kanatsoulis[*], *Xiao Fu*[†], *Nicholas D. Sidiropoulos*[⋆], *and Mingyi Hong*[*]

[*]Department of ECE, University of Minnesota, Minneapolis, MN, USA
[†]School of EECS, Oregon State University, Corvallis, OR, USA
[⋆]Department of ECE, University of Virginia, Charllotesville, VA, USA

ABSTRACT

The *sum-of-correlations* (SUMCOR) *generalized canonical correlation analysis* (GCCA) aims at producing low-dimensional representations of multiview data via enforcing pairwise similarity of the reduced-dimension views. SUMCOR has been applied to a large variety of applications including blind separation, multilingual word embedding, and cross-modality retrieval. Despite the NP-hardness of SUMCOR, recent work has proposed effective algorithms for handling it at very large scale. However, the existing scalable algorithms are not easy to extend to incorporate structural regularization and prior information – which are critical for real-world applications where outliers and modeling mismatches are present. In this work, we propose a new computational framework for large-scale SUMCOR GCCA. The algorithm can easily incorporate a suite of structural regularizers which are frequently used in data analytics, has lightweight updates and low memory complexity, and can be easily implemented in a parallel fashion. The proposed algorithm is also guaranteed to converge to a Karush-Kuhn-Tucker (KKT) point of the regularized SUMCOR problem. Carefully designed simulations are employed to demonstrate the effectiveness of the proposed algorithm.

***Index Terms*—** Generalized canonical correlation analysis, SUMCOR, multi-view analysis, regularization, feature extraction

1. INTRODUCTION

Canonical correlation analysis (CCA) is a classic analytical tool [1–3] which has a wide spectrum of applications in signal processing and machine learning. CCA aims at extracting common low-dimensional structure of the same set of entities measured in different high-dimensional feature spaces (also called 'views' or 'modalities'), e.g., image and speech of a person. In recent years, CCA has been successfully applied to a variety of domains including brain imaging [4], blind source separation [5], speech recognition [6], and word embedding [7, 8].

Despite the nonconvex appearance of the classic two-view CCA problem, it can be converted to a generalized eigendecomposition problem and solved rather efficiently [9]. In recent years, there has been renewed interest in scaling up solvers for two-view CCA. The proposed scalable algorithms are mostly judicious ways of computing the generalized eigendecomposition problem; see [10–13]. On the other hand, extensions to the so-called *generalized canonical correlation analysis* (GCCA) that considers multiple (more than two) views are highly nontrivial. The arguably most natural extension of the two-view CCA is the sum-of-correlations (SUMCOR) GCCA [2, 3, 14–18]. SUMCOR looks for common structure of multiple views via enforcing pairwise similarities between the reduced-dimension views. Unlike the two-view CCA, SUMCOR has been shown to be NP-hard [18]. In the literature, many algorithms such as alternating optimization (AO) [14], modified power method [15], and semidefinite relaxation [18] have been introduced to handle the SUMCOR problem. However, most of them have serious complexity issues when the views are large (e.g., when dealing with views of size $10,000 \times 10,000$, the AO algorithm in [14] will create large whitening matrices occupying 35GB of memory). To address this issue, Fu et al. [7] proposed a new computational framework and substantially scaled up SUMCOR, by exploiting data sparsity.

The algorithms proposed in [7] are scalable and effective. There are, however, important challenges remaining. The algorithms in [7] are based on a delicate change of variables that hinges on the specific formulation of the original SUMCOR optimization problem. However, in practice, modifications to the SUMCOR formulation are often needed. For example, in many cases one knows that some features of the views are irrelevant or even damaging to the correlation seeking process (e.g., 'stop words' in text analytics, and non-informative genes in genetics). In such cases, sparsity regularizers have been considered to preclude such irrelevant features when performing CCA/GCCA [8, 19–21]. Unfortunately, the existing sparse GCCA algorithms are mostly designed for small/medium-scale problems and thus are not suitable for big data analytics.

In this work, we propose a new computational framework to handle the large-scale regularized SUMCOR problem. Our idea is to employ a penalty-dual decomposition (PDD) technique to 'split' the effort of tackling the already very hard manifold constraints of SUMCOR and the newly added regularizers. Our approach has an array of desired features. First, the algorithm admits lightweight updates leveraging data sparsity. In addition, a variety of regularizers that are frequently used in data analytics, such as sparsity, group sparsity, elastic net, smoothness, and nonnegativity can be easily handled by our new framework. Third, the updates can be naturally distributed to different computational agents with limited communication overhead, which can further reduce the overall runtime. Last, the proposed algorithm is guaranteed to converge to a Karush-Kuhn-Tucker (KKT) point of the problem of interest.

Note that the regularized SUMCOR has nonconvex constraints, and thus commonly used primal-dual methods in general do not provide convergence guarantees. Nevertheless, with carefully designed updating rules that leverage recent theoretical results on generic PDD algorithms [22], convergence can be shown. Simulations show that the algorithm is promising for handling big multiview data and promoting the imposition of structural properties on the sought canonical components.

2. PROBLEM STATEMENT

Let us consider a two-view data set, where $\boldsymbol{Z}_1 \in \mathbb{R}^{L \times M_1}$ and $\boldsymbol{Z}_2 \in \mathbb{R}^{L \times M_2}$ are the two views – i.e., $\boldsymbol{Z}_1(\ell,:) \in \mathbb{R}^{1 \times M_1}$ and $\boldsymbol{Z}_2(\ell,:) \in$

This work was supported in part by National Science Foundation under Project NSF ECCS-1608961 and Project NSF IIS-1447788.

$\mathbb{R}^{1\times M_2}$ are two high-dimensional representations of the entity ℓ in two different feature spaces (e.g., speech and image of a person). Let us assume that $\boldsymbol{X}_i = (1/\sqrt{L})(\boldsymbol{Z}_i - \mathbf{1}\boldsymbol{d}_i^T)$ is the scaled and centered version of the ith view, where $\boldsymbol{d}_i^T = (1/L)\sum_{\ell=1}^{L}\boldsymbol{Z}(\ell,:)$ is the sample mean of the ith view. The classic two-view CCA can be expressed in the following optimization form [12, 17, 23, 24]:

CCA:

$$\underset{\boldsymbol{Q}_1,\boldsymbol{Q}_2}{\text{maximize}}\ \text{Tr}\left(\boldsymbol{Q}_1^T\boldsymbol{X}_1^T\boldsymbol{X}_2\boldsymbol{Q}_2\right) \tag{1a}$$

$$\text{subject to}\ \boldsymbol{Q}_i^T\left(\boldsymbol{X}_i^T\boldsymbol{X}_i\right)\boldsymbol{Q}_i = \boldsymbol{I}_K,\ i=1,2, \tag{1b}$$

where $\boldsymbol{I}_K$ denotes a $K\times K$ identity matrix, $\boldsymbol{Q}_i\in\mathbb{R}^{M_i\times K}$ denotes a dimensionality-reducing matrix of view i, K is the number of canonical components that we seek and $K \ll \min\{M_i, L\}$ in big data analytics. Maximizing $\text{Tr}(\boldsymbol{Q}_1^T\boldsymbol{X}_1^T\boldsymbol{X}_2\boldsymbol{Q}_2)$ subject to the constraints in (1b) is equivalent to maximizing the cross-correlation between the reduced-dimension views $\boldsymbol{X}_1\boldsymbol{Q}_1$ and $\boldsymbol{X}_2\boldsymbol{Q}_2$.

The arguably most natural extension of the two-view CCA is the sum-of-correlations (SUMCOR) GCCA [2, 3, 14–18].

SUMCOR GCCA:

$$\underset{\{\boldsymbol{Q}_i\}_{i=1}^I}{\text{maximize}}\ \sum_{i=1}^{I}\sum_{j>i}^{I}\text{Tr}\left(\boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_j\boldsymbol{Q}_j\right) \tag{2}$$

$$\text{subject to}\ \boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_i\boldsymbol{Q}_i = \boldsymbol{I}_K,\ i=1,\ldots,I.$$

where $I>2$ is the number of available views. SUMCOR looks for common structure among multiple views by enforcing pairwise similarities between the reduced-dimension views, i.e., $\boldsymbol{X}_i\boldsymbol{Q}_i$'s. Unlike the two-view CCA whose optimal solution amounts to an eigendecomposition, SUMCOR is NP-hard [15, 18]. Nevertheless, effective and scalable solvers for SUMCOR exist – see the recent work in [7].

In many practical cases, the plain GCCA/CCA in (1) and (2) are not enough to learn closely related latent representations of the views. This is because there is always noise and model mismatches in practice – which could severely hinder the ability of GCCA/CCA to extract common information from multiple views. To circumvent this situation, using prior and structural information about the sought canonical components has been considered to assist GCCA/CCA, resulting in the regularized GCCA formulation:

Regularized SUMCOR GCCA:

$$\underset{\{\boldsymbol{Q}_i\}_{i=1}^I}{\text{minimize}}\ -\sum_{i=1}^{I}\sum_{j>i}^{I}\text{Tr}\left(\boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_j\boldsymbol{Q}_j\right) + \sum_{i=1}^{I}\lambda_i r_i(\boldsymbol{Q}_i)$$

$$\text{subject to}\ \boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_i\boldsymbol{Q}_i = \boldsymbol{I}_K,\ i=1,\ldots,I, \tag{3}$$

where $\lambda_i\geq 0$ is a regularization parameter which strikes a balance between the SUMCOR objective and structure-promotion. Several regularization terms are of particular interest. For example,

$$\|\boldsymbol{Q}_i\|_1 = \sum_{m=1}^{M_i}\sum_{k=1}^{K}|\boldsymbol{Q}_i(m,k)|\ \ \&\ \|\boldsymbol{Q}_i\|_{2,1} = \sum_{m=1}^{M_i}\|\boldsymbol{Q}_i(m,:)\|_2,$$

are often used in the literature as $r_i(\boldsymbol{Q}_i)$ for feature selection [8, 21]. Taking $\|\boldsymbol{Q}_i\|_{2,1}$ as an example, it is known that the $\ell_{2,1}$-norm promotes row-sparsity, and zero rows in $\boldsymbol{Q}_i$ will nullify corresponding columns in $\boldsymbol{X}_i$ when forming $\boldsymbol{X}_i\boldsymbol{Q}_i$. This is very effective in suppressing irrelevant features and outliers in $\boldsymbol{X}_i$. There are also other regularization terms, such as minimum energy and elastic net, which are frequently used for different purposes – see [8].

Note that the manifold constraints in (3) together with the regularization term make the optimization problem very challenging. The algorithm in [14] approximates Problem (3) using a similar formulation while relaxing the constraint to be $\|\boldsymbol{q}_i\|_2\leq 1$ for the $K=1$ case, and then finds other components of $\boldsymbol{Q}_i$ through a deflation procedure. Changing the constraints is apparently undesired, and deflation is prone to error propagation. In [20], an alternating direction method of multiplier (ADMM) procedure was proposed to handle the two-view case with sparsity regularization. The procedure and proof are tailored for the two-view case and cannot cover the general SUMCOR case. More importantly, the algorithm there involves forming the term $(\boldsymbol{X}_i^T\boldsymbol{X}_i)^{-1/2}$, which is a *dense* $M_i\times M_i$ matrix even when $\boldsymbol{X}_i$ is sparse – if $M_i = 10,000$, this matrix costs 35 GB memory when double precision is employed. In addition, computing the inverse of $(\boldsymbol{X}_i\boldsymbol{X}_i)^{1/2}$ consumes $\mathcal{O}(10^{12})$ flops, which is also too costly for big data analytics. In [8], a regularized large-scale algorithm was proposed for the MAX-VAR formulation of GCCA. However, the algorithm cannot be extended to cover the SUMCOR case.

3. PROPOSED ALGORITHM

In this section, we address the challenging optimization problem of large-scale regularized SUMCOR GCCA. Specifically, we will propose an algorithmic framework that is able to handle huge multiview data under a variety of structural regularizers on the canonical components – with affordable memory and computational costs. The proposed framework can also easily facilitate parallel computations with limited communication overhead.

To begin with, let us consider the alternative formulation of structured GCCA:

$$\underset{\{\boldsymbol{Q}_i\}_{i=1}^I}{\text{minimize}}\ \sum_{i=1}^{I}\sum_{j>i}^{I}\frac{1}{2}\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{X}_j\boldsymbol{Q}_j\|_F^2 + \sum_{i=1}^{I}\lambda_i r_i(\boldsymbol{Q}_i) \tag{4}$$

$$\text{subject to}\ \boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_i\boldsymbol{Q}_i = \boldsymbol{I}_K,\ i=1,\ldots,I.$$

Note that if one expands the first term in the objective and discards the constants, the formulation in (3) is recovered. To proceed, we re-write the above as

$$\underset{\{\boldsymbol{Q}_i,\boldsymbol{G}_i\}_{i=1}^I}{\text{minimize}}\ \sum_{i=1}^{I}\sum_{j>i}^{J}\frac{1}{2}\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}_j\|_F^2 + \sum_{i=1}^{I}\lambda_i r_i(\boldsymbol{Q}_i) \tag{5a}$$

$$\text{subject to}\ \boldsymbol{G}_i = \boldsymbol{X}_i\boldsymbol{Q}_i,\ \boldsymbol{G}_i^T\boldsymbol{G}_i = \boldsymbol{I}_K,\ \forall i, \tag{5b}$$

where the slack variable $\boldsymbol{G}_i\in\mathbb{R}^{L\times K}$ is a thin matrix. This way, we have not changed the optimization problem, but 'split the challenge' using different variables – since we do not wish to handle the regularization terms (which are usually nonsmooth) and the manifold constraints together. To deal with Problem (5), we propose to employ a primal-dual approach. Specifically, we consider the augmented Lagrangian of Problem (5), which is

$$\mathcal{L}\left(\{\boldsymbol{Q}_i,\boldsymbol{G}_i,\boldsymbol{Y}_i\}_{i=1}^I\right) = \sum_{i=1}^{I}\sum_{j>i}^{J}\frac{1}{2}\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}_j\|_F^2 + \sum_{i=1}^{I}\lambda_i r_i(\boldsymbol{Q}_i)$$
$$+\frac{\rho}{2}\sum_{i=1}^{I}\left\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}_i + \frac{1}{\rho}\boldsymbol{Y}_i\right\|_F^2,$$

where $\boldsymbol{Y}_i$ is the dual variable associated with the equality constraint $\boldsymbol{X}_i\boldsymbol{Q}_i = \boldsymbol{G}_i$. At this point, it is tempting to apply ADMM to handle the augmented Lagrangian, since one can see that the subproblems

w.r.t. $\boldsymbol{Q}_i$, $\boldsymbol{G}_i$ and $\boldsymbol{Y}_i$ are easy to solve under the ADMM framework [25]. However, ADMM is not guaranteed to converge when nonconvex constraints are involved. Here, we propose to employ a delicately modified version of ADMM, namely, the penalty-dual decomposition (PDD) framework [22]. The proposed algorithm is presented in Algorithm 1. The PDD algorithm consists of two modules. The first module is a sub-solver (cf. line 3 in Algorithm 1) that handles the augmented Lagrangian w.r.t. $\{\boldsymbol{Q}_i\}$ and $\{\boldsymbol{G}_i\}$ when $\boldsymbol{Y}_i$'s are fixed. The second module of PDD makes a decision on updating the dual variables or the penalty parameter $\rho^{(r)}$ (cf. lines 4-8 in Algorithm 1). In the sub-solver, the parameter $\epsilon^{(r)}$ is for specifying the accuracy of the sub-solver solution at iteration r. To be specific, the sub-solver aims at solving the following problem:

$$\underset{\{\boldsymbol{Q}_i,\boldsymbol{G}_i:\boldsymbol{G}_i^T\boldsymbol{G}_i=\boldsymbol{I}_K\}}{\text{minimize}} \ \mathcal{L}\left(\{\boldsymbol{Q}_i,\boldsymbol{G}_i,\boldsymbol{Y}_i\}_{i=1}^I\right) \tag{6}$$

Note that the solver does not need to solve the above to optimality. Every call of the sub-solver only requires that $\boldsymbol{Q}_i$ and $\boldsymbol{G}_i$ converge to a neighborhood of a KKT point of (6), roughly speaking, where the 'diameter' of the neighborhood is specified by $\epsilon^{(r)}$ – see Eq. (8).

One easily implementable solver for (6) is the so-called inexact alternating optimization [26]. Specifically, one may employ the following updates alternately between $\{\boldsymbol{Q}_i\}$ and $\{\boldsymbol{G}_i\}$:

$$\boldsymbol{Q}_i^+ \leftarrow \arg\min_{\boldsymbol{Q}_i} \ \left\|\boldsymbol{Q}_i - (\hat{\boldsymbol{Q}}_i - \alpha\nabla f(\hat{\boldsymbol{Q}}_i))\right\|_F^2 + \lambda_i r_i(\boldsymbol{Q}_i), \tag{7a}$$

$$\boldsymbol{G}_i^+ \leftarrow \arg\min_{\boldsymbol{G}_i^T\boldsymbol{G}_i=\boldsymbol{I}_K} \sum_{j=1,j\neq i}^{J} \frac{1}{2}\left\|\boldsymbol{X}_j\boldsymbol{Q}_j^+ - \boldsymbol{G}_i\right\|_F^2, + \frac{\rho^{(r)}}{2}\left\|\boldsymbol{X}_i\boldsymbol{Q}_i^+ - \boldsymbol{G}_i + \frac{1}{\rho^{(r)}}\boldsymbol{Y}_i\right\|_F^2 \tag{7b}$$

$$\hat{\boldsymbol{Q}}_i \leftarrow \boldsymbol{Q}_i^+, \ \hat{\boldsymbol{G}}_i \leftarrow \boldsymbol{G}_i^+. \tag{7c}$$

where $\hat{\boldsymbol{Q}}_i$ and $\boldsymbol{Q}_i^+$ are the old and new iterates of $\boldsymbol{Q}_i$, and $\nabla f(\hat{\boldsymbol{Q}}_i) = \sum_{j=1,j>i}^{I}(\boldsymbol{X}_i^T\boldsymbol{X}_i\hat{\boldsymbol{Q}}_i - \boldsymbol{X}_i^T\hat{\boldsymbol{G}}_j) + \rho^{(r)}(\boldsymbol{X}_i^T\boldsymbol{X}_i\hat{\boldsymbol{Q}}_i - \boldsymbol{X}_i^T(\hat{\boldsymbol{G}}_i - 1/\rho^{(r)}\boldsymbol{Y}_i))$ is the partial gradient with respect to $\boldsymbol{Q}_i$ of the smooth part of the objective in (5).

There are many favorable features of the algorithm: First, the updates of the $\boldsymbol{Q}_i$'s and $\boldsymbol{G}_i$'s can be very lightweight if the views are sparse. A complexity order of $\mathcal{O}(\text{nnz}(\boldsymbol{X}_i)K)$ flops per iteration ($\text{nnz}(\boldsymbol{X})$ = number of non-zeros in $\boldsymbol{X}$) is enough to compute the gradient $\nabla f(\hat{\boldsymbol{Q}}_i)$, since multiplications such as $\boldsymbol{X}_i^T\boldsymbol{G}_i$, $\boldsymbol{X}_i\boldsymbol{Q}_i$ and $\boldsymbol{X}_i^T(\boldsymbol{X}_i\boldsymbol{Q}_i)$ all consume $\mathcal{O}(\text{nnz}(\boldsymbol{X}_i)K)$ flops. Then, solving (7a) amounts to a proximal operator, whose complexity is linear in the size of $\boldsymbol{Q}_i$, if $r_i(\boldsymbol{Q}_i)$ is some proximity-friendly function such as ℓ_1-norm, $\ell_{2,1}$-norm, and elastic net [27]. The subproblem in (7b) can be solved by economy-size SVD, which only needs $\mathcal{O}(LK^2)$ flops to carry out. Second, the update of $\boldsymbol{Y}_i$ also costs very few flops since $\boldsymbol{X}_i\boldsymbol{Q}_i$ has already been computed and only additions of thin matrices are left. Third, all the subproblems cost $\mathcal{O}(LK)$ memory – which is very cheap. Furthermore, the algorithmic structure is friendly for parallel computing; i.e., $\boldsymbol{Q}_i$ and $\boldsymbol{G}_i$ for $i = 1, \ldots, I$ can be updated simultaneously at different computing agents, respectively. What need to be exchanged among the agents are $\boldsymbol{G}_i$ and $\boldsymbol{Y}_i$, which are merely 'thin matrices' of size $L \times K$ and thus do not cost much communication overhead.

At a high level, PDD can be considered as a variant of ADMM, which also aims at solving the augmented Lagrangian while changing the weight of the penalty, i.e., ρ, along the iterations. The ρ parameter and the dual variables both help enforce the lifted equality constraints, depending on the 'level of violation' at a particular iteration. If the equality constraint is heavily violated, the algorithm increases ρ so that the next iteration will put more emphasis on the penalty. This way, and along with a delicately designed updating order of the primal and dual variables, convergence of the algorithm can be guaranteed even with nonconvex constraints involved. Regarding convergence properties of the PDD based GCCA, we show that:

Algorithm 1: `PDD-GCCA`

input : $\{\boldsymbol{X}_i\}_{i=1}^I$; K; $\rho^{(0)} > 0$; $0 < c < 1$; $\{\epsilon^{(r)}, \eta^{(r)}\}_{r=1}^{\infty}$.

1 $r \leftarrow 0$;
2 **repeat**
3 $\quad \left(\{\boldsymbol{Q}_i^{(r+1)}, \boldsymbol{G}_i^{(r+1)}\}_{i=1}^I\right) \leftarrow$ `sub-solver` $\left(\{\boldsymbol{Y}_i^{(r)}\}_i, \rho^{(r)}, \epsilon^{(r)}\right)$ for (6);
4 $\quad$ **if** $\sum_{i=1}^I \|\boldsymbol{X}_i\boldsymbol{Q}_i^{(r+1)} - \boldsymbol{G}_i^{(r+1)}\|_F^2 \leq \eta^{(r)}$ **then**
5 $\quad\quad \boldsymbol{Y}_i^{(r+1)} = \boldsymbol{Y}_i^{(r)} + \rho^{(r)}(\boldsymbol{X}\boldsymbol{Q}_i^{(r+1)} - \boldsymbol{G}_i^{(r+1)})$, $\rho^{(r+1)} = \rho^{(r)}$;
6 $\quad$ **else**
7 $\quad\quad \boldsymbol{Y}_i^{(r+1)} = \boldsymbol{Y}_i^{(r)}$, $\rho^{(r+1)} = c\rho^{(r)}$;
8 $\quad$ **end**
9 $\quad r \leftarrow r + 1$;
10 **until** *some stopping criterion is reached*;

output: $\{\boldsymbol{G}_i\}$

Proposition 1 *Assume that $\epsilon^{(r)} \to 0$, $\eta^{(r)} \to 0$ as $r \to \infty$ and that the stopping criterion of the* `sub-solver` *involved in Algorithm 1 is*

$$\max\left(\|\hat{\boldsymbol{Q}}_i - \boldsymbol{Q}_i^+\|_\infty, \|\hat{\boldsymbol{G}}_i - \boldsymbol{G}_i^+\|_\infty\right) \leq \epsilon^{(r)}, \quad \forall r. \tag{8}$$

Then, every limit point of the solution sequence produced by the proposed `PDD-GCCA` *algorithm is a KKT point of Problem* (4)*.*

Due to the space limitation, we only present the main ideas behind the proof here, and relegate the complete proof to the forthcoming journal version.

Proof Sketch: The proof consists of three parts. First, we show that the sub-solver produces KKT points of the subproblem (6). This part can be shown by applying a similar technique as in [8, 28–30]. Then, we show that the KKT points of Problem (3) satisfy the so-called Robinson condition [22, 31]. The final step is to apply the result for generic penalty-dual decomposition [22] to show that the proposed algorithm converges to a KKT point. We should mention that the first difficulty of the proof lies in convergence of the sub-solver, since it was not obvious whether or not the updates in (7a)-(7b) reach a KKT point of Probem (6) – the problem has nonconvex constraints and applying inexact alternating optimization is not guaranteed to converge by simply invoking existing analyses such as those in [26]; tailored analysis is needed for this step. Another key challenge is to verify the Robinson's condition, which is not easily doable in general cases. Fortunately, the regularized SUMCOR problem can be verified to satisfy this condition.

Another comment is that, in theory, the sub-solver is required to satisfy increasingly stringent stopping criteria as specified in Proposition 1 to guarantee convergence. In practice, we observe that this is not necessary. Using a fixed number of iterations to implement the sub-solver usually suffices to offer quick convergence – and this is consistent with the observation in [22], where the PDD framework was applied to a number of different problems.

4. SIMULATIONS

In this section we showcase the effectiveness of PDD-GCCA using large-scale simulations. The multiple views are generated as follows. We assume that the views share a common latent factor $\boldsymbol{S} \in \mathbb{R}^{L\times M_i}$

which is a randomly generated sparse matrix whose non-zero entries follow the zero-mean unit-variance Gaussian distribution. Each view $\boldsymbol{X}_i$ is generated following $\boldsymbol{X}_i = \boldsymbol{S}\boldsymbol{A}_i$, where $\boldsymbol{A}_i \in \mathbb{R}^{M_i \times M_i}$ is a matrix that maps the shared factor to the ith view. The density level of $\boldsymbol{S}$ and $\boldsymbol{A}_i$ is controlled such that the density level of each view, i.e., $\texttt{density}_i = \frac{\texttt{nnz}(\mathbf{X}_i)}{LM_i}$, is controlled. The number of features of each view is set to be equal for simplicity, i.e. $M_i = M$.

We first test the algorithms when the views do not have outlying features. Under such cases, we let $r_i(\boldsymbol{Q}_i) = 0$ and (4) recovers the classic SUMCOR GCCA formulation. For such cases, we adopt the recently proposed large-scale SUMCOR algorithms *LasCCA* and *DisCCA* [7] as baselines; LasCCA and DisCCA were shown to be state-of-the-art when handling very large-scale and sparse mutiview data [7]. To evaluate the performance, we observe the total correlation captured. Since the views share a common latent factor, they are perfectly correlated in the shared latent domain. Therefore, the optimal value of total correlation is achieved when $\boldsymbol{A}_i\boldsymbol{Q}_i$'s are perfectly aligned with each other – which yield a SUMCOR value of $KI(I-1)$. We also record the time each algorithm needs to capture 95% of the optimal correlation, denoted as `95%-time`. The number of observations is $L = 120,000$ and each view is defined by an $M = 100,000$ features. The number of canonical components is set to $K = 5$. The parameter $\rho^{(0)}$ for PDD-GCCA is set to be 2. The maximal number of iterations of the sub-solver is set to be 5. The sequence for the primal residual is chosen to be $\eta^{(r)} = \frac{100}{r}$ and $c = 0.9$. The results are averaged over 20 Monte Carlo trials.

Table 1 shows the performance of the algorithms under various density levels. One can see that LasCCA works very well in terms of capturing correlations. Under all the tested density levels, LasCCA captures more than 98% of the total SUMCOR (which is 100 in this simulation). DisCCA also works well when $\texttt{density} = 5 \times 10^{-5}$ and 10^{-4}, but less competitive when $\texttt{density} = 10^{-5}$. PDD-GCCA has comparable performance relative to LasCCA in terms of capturing correlations, but PDD-GCCA works much faster – when using a single core implementation, it is already at least 3 times faster than the benchmarking algorithms. This may be because PDD-GCCA uses a primal-dual framework, and dual updates usually help expedite convergence – while both LasCCA and DisCCA are primal algorithms which do not employ any dual updates. In addition, if one employs a multicore implementation of PDD-GCCA (recall that PDD-GCCA can be implemented distributively using I cores), the runtime performance of PDD-GCCA is even better – as a result, the multicore version is at around 10 times faster than the baselines for capturing 95% of the total correlations.

Table 1: Evaluation of the algorithms; PDD-GCCA uses $r_i(\cdot) = 0$; $L = 120,000$ and $M = 100,000$.

Algorithm	metric	density level		
		10^{-4}	5×10^{-5}	10^{-5}
PDD-GCCA (multicore)	corr. captured	99.66	**99.59**	**99.79**
	95% time (sec)	**6.82**	**6.08**	**7.43**
PDD-GCCA	corr. captured	**99.67**	**99.59**	**99.79**
	95% time (sec)	22.06	14.64	18.57
LasCCA	corr. captured	98.96	98.77	99.37
	95% time (sec)	59.56	63.23	87.5
DisCCA (multicore)	corr. captured	96.78	95.61	78.62
	95% time (sec)	54.24	71.37	inf
DisCCA	corr. captured	96.78	95.61	78.62
	95% time (sec)	133.20	144.97	inf

We also test the algorithms when outlying features are present in the views. To simulate such scenarios the data are generated as follows: $\boldsymbol{X}_i = [\boldsymbol{S}\boldsymbol{A}_i, \boldsymbol{O}_i] + \boldsymbol{N}$, where $\boldsymbol{S}$ and $\boldsymbol{A}_i$ are defined as before, $\boldsymbol{N}$ is zero mean 0.01 variance sparse Gaussian noise and $\boldsymbol{O}_i \in \mathbb{R}^{L \times M_o}$ is also a sparse matrix with zero-mean and unit variance non zero entries – but completely uncorrelated across the views. The 'signal part' and the 'outlier' part of each view are enforced to have comparable energy levels, i.e. $\|\boldsymbol{S}\boldsymbol{A}_i\|_F \approxeq \|\boldsymbol{O}_i\|_F$, so that simple energy detection could not identify the outlying features. In this case, GCCA has two objectives. The first is to capture the highest possible correlation between the informative part of the views, while at the same time to suppress the impact of $\boldsymbol{O}_i$. Towards this end, the $\ell_{2,1}$ and ℓ_1 norms that promote (row-)sparsity are employed to serve as $r_i(\boldsymbol{Q}_i)$. In order to evaluate the performance of the algorithms, two metrics are introduced as in [8]. Let $\mathcal{I}_s$ and $\mathcal{I}_o$ be the index sets of the signal and the outlying columns in $\boldsymbol{X}_i$, respectively, where $\mathcal{I}_o \bigcup \mathcal{I}_s = \{1, \ldots, L\}$. The first metric is: $\texttt{metric\_1} = \sum_{i=1}^{I}\sum_{j>i}^{I} \text{Tr}\left(\boldsymbol{Q}_i(\mathcal{I}_s,:)^T\boldsymbol{X}_i(:,\mathcal{I}_s)^T\boldsymbol{X}_j(:,\mathcal{I}_s)\boldsymbol{Q}_j(\mathcal{I}_s,:)\right)\frac{100}{KI(I-1)}$, which measures the percentage of total signal correlation captured. In our simulations, $\texttt{metric\_1} \in [0, 100]$, and a higher value of `metric_1` is desired. The second metric measures the ability of identifying and suppressing the outlying part. To this end, we define $\texttt{metric\_2} = \sum_{i=1}^{I}\|\boldsymbol{Q}_i(\mathcal{I}_o,:)\|_F$ whose optimal value is zero and smaller values of `metric_2` correspond to better performance in suppressing outliers.

Table 2 shows the performance of the algorithms. The number of observations is $L = 100,000$ that contain $80,000$ informative and $80,000$ outlying features. The number of canonical components varies from $K = 5$ to $K = 50$. The density level of each view is $\texttt{density} = 10^{-4}$. The regularization parameter λ_i is chosen to be 0.1. The results are averaged over 20 Monte Carlo simulations as before. One can see that PDD-GCCA with the $\ell_{2,1}$ norm regularization remarkably outperforms the unregularized ones: it captures more than 92% of the total signal correlation in all cases and it successfully suppresses the outlying features according to the values of `metric_2`. PDD-GCCA with ℓ_1 regularizers also works well. This simulation also justifies our motivation of considering regularized GCCA: When outlying features are present, classic GCCA may not be able to produce satisfactory results.

Table 2: Performance of the algorithms in the presence of outliers.

Algorithm	metric	# of canonical components			
		$K = 5$	$K = 10$	$K = 20$	$K = 50$
PDD-GCCA	metric 1	27.78	28.44	29.21	29.24
	metric 2	2.99	4.18	5.81	9.13
PDD-GCCA (ℓ_1)	metric 1	91.14	**92.57**	91.74	92.13
	metric 2	**0.55**	**0.88**	1.35	2.30
PDD-GCCA ($\ell_{2,1}$)	metric 1	**92.26**	92.43	**94.72**	**95.83**
	metric 2	0.63	1.03	**1.21**	**1.86**
LasCCA	metric 1	37.61	40.88	39.59	38.37
	metric 2	2.51	3.41	4.84	7.66
DisCCA	metric 1	18.28	18.44	17.61	17.27
	metric 2	3.49	4.93	7.11	11.16

5. CONCLUSION

In this work, the regularized SUMCOR GCCA problem has been considered. A scalable algorithm that is based on penalty-dual decomposition has been proposed to address this challenging optimization problem. The proposed PDD-GCCA algorithm can easily incorporate many different regularizers to enforce structural canonical components, and thus is very flexible. It also admits lightweight updates and low memory complexity when handling large sparse data. The proposed algorithmic framework is friendly to parallel computing – the variable splitting and dual decomposition nature of the algorithmic structure can easily facilitate distributed implementation with limited communication overhead. Despite the hardness of analyzing convergence properties of primal-dual optimization involving nonconvex constraints, the algorithm features KKT convergence assurance. Simulations on synthetic large-scale data have been employed to demonstrate the effectiveness of the proposed algorithm.

6. REFERENCES

[1] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[2] J. R. Kettenring, "Canonical analysis of several sets of variables," *Biometrika*, vol. 58, no. 3, pp. 433–451, 1971.

[3] J. D. Carroll, "Generalization of canonical correlation analysis to three or more sets of variables," in *Proceedings of the 76th annual convention of the American Psychological Association*, vol. 3, 1968, pp. 227–228.

[4] J. Sui, T. Adali, Q. Yu, J. Chen, and V. D. Calhoun, "A review of multivariate methods for multimodal fusion of brain imaging data," *Journal of neuroscience methods*, vol. 204, no. 1, pp. 68–81, 2012.

[5] Y.-O. Li, T. Adali, W. Wang, and V. D. Calhoun, "Joint blind source separation by multiset canonical correlation analysis," *IEEE Transactions on Signal Processing*, vol. 57, no. 10, pp. 3918–3929, 2009.

[6] R. Arora and K. Livescu, "Multi-view learning with supervision for transformed bottleneck features," in *Proc. ICASSP 2014*, 2014, pp. 2499–2503.

[7] X. Fu, K. Huang, E. Papalexakis, H. Song, P. Talukdar, N. D. Sidiropoulos, C. Faloutsos, and T. Mitchell, "Efficient and distributed algorithms for large-scale generalized correlation analysis," in *Proc. ICDM 2016*. IEEE, 2016.

[8] X. Fu, K. Huang, M. Hong, N. D. Sidiropoulos, and A. M.-C. So, "Scalable and flexible max-var generalized canonical correlation analysis via alternating optimization," *IEEE Trans. Signal Process.,* to appear, 2017.

[9] G. H. Golub and C. F. V. Loan., *Matrix Computations*. The Johns Hopkins University Press, 1996.

[10] R. Ge, C. Jin, S. M. Kakade, P. Netrapalli, and A. Sidford, "Efficient algorithms for large-scale generalized eigenvector computation and canonical correlation analysis," *arXiv preprint arXiv:1604.03930*, 2016.

[11] Z. Allen-Zhu and Y. Li, "Doubly accelerated methods for faster cca and generalized eigendecomposition," *arXiv preprint arXiv:1607.06017*, 2016.

[12] Y. Lu and D. P. Foster, "Large scale canonical correlation analysis with iterative least squares," in *NIPS*, 2014, pp. 91–99.

[13] W. Wang, J. Wang, and N. Srebro, "Globally convergent stochastic optimization for canonical correlation analysis," *arXiv preprint arXiv:1604.01870*, 2016.

[14] A. Tenenhaus and M. Tenenhaus, "Regularized generalized canonical correlation analysis," *Psychometrika*, vol. 76, no. 2, pp. 257–284, 2011.

[15] L.-H. Zhang, L.-Z. Liao, and L.-M. Sun, "Towards the global solution of the maximal correlation problem," *Journal of Global Optimization*, vol. 49, no. 1, pp. 91–107, 2011.

[16] M. T. Chu and J. L. Watterson, "On a multivariate eigenvalue problem, part i: Algebraic theory and a power method," *SIAM Journal on scientific computing*, vol. 14, no. 5, pp. 1089–1106, 1993.

[17] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural computation*, vol. 16, no. 12, pp. 2639–2664, 2004.

[18] J. Rupnik, P. Skraba, J. Shawe-Taylor, and S. Guettes, "A comparison of relaxations of multiset cannonical correlation analysis and applications," *arXiv preprint arXiv:1302.0974*, 2013.

[19] D. Chu, L.-Z. Liao, M. K. Ng, and X. Zhang, "Sparse canonical correlation analysis: new formulation and algorithm," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 12, pp. 3050–3065, 2013.

[20] C. Gao, Z. Ma, and H. H. Zhou, "An efficient and optimal method for sparse canonical correlation analysis," *arXiv preprint arXiv:1409.8565*, 2014.

[21] D. M. Witten and R. J. Tibshirani, "Extensions of sparse canonical correlation analysis with applications to genomic data," *Statistical applications in genetics and molecular biology*, vol. 8, no. 1, pp. 1–27, 2009.

[22] Q. Shi and M. Hong, "Penalty dual decomposition method with application in signal processing," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4059–4063.

[23] G. H. Golub and H. Zha, *The canonical correlations of matrix pairs and their numerical computation*. Springer, 1995.

[24] Z. Ma, Y. Lu, and D. Foster, "Finding linear structure in large datasets with scalable canonical correlation analysis," in *ICML 2015*, 2015.

[25] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, pp. 1–122, 2011.

[26] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, 2013.

[27] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in optimization*, vol. 1, no. 3, pp. 123–231, 2013.

[28] J. Tranter, N. D. Sidiropoulos, X. Fu, and A. Swami, "Fast unit-modulus least squares with applications in beamforming," *IEEE Transactions on Signal Processing*, vol. 65, no. 11, pp. 2875–2887, 2017.

[29] X. Fu, W.-K. Ma, K. Huang, and N. D. Sidiropoulos, "Blind separation of quasi-stationary sources: Exploiting convex geometry in covariance domain," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2306–2320, May 2015.

[30] C. Qian, X. Fu, N. D. Sidiropoulos, L. Huang, and J. Xie, "Inexact alternating optimization for phase retrieval in the presence of outliers," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 6069–6082, Nov 2017.

[31] A. P. Ruszczyński, *Nonlinear optimization*. Princeton university press, 2006, vol. 13.