SCALABLE AND FLEXIBLE MAX-VAR GENERALIZED CANONICAL CORRELATION ANALYSIS VIA ALTERNATING OPTIMIZATION

Xiao Fu, Kejun Huang*, Mingyi Hong⋆, Nicholas D. Sidiropoulos*, Anthony Man-Cho So†*

*Department of ECE, University of Minnesota, Minneapolis, MN, USA
⋆Department of IME, Iowa State University, Ames, IA, USA
†Department of SEEM, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong

ABSTRACT

Unlike dimensionality reduction (DR) tools for single-view data, e.g., principal component analysis (PCA), canonical correlation analysis (CCA) and generalized CCA (GCCA) are able to integrate information from multiple feature spaces of data. This is critical in multi-modal data fusion and analytics, where samples from a single view may not be enough for meaningful DR. In this work, we focus on a popular formulation of GCCA, namely, MAX-VAR GCCA. The classic MAX-VAR problem is optimally solvable via eigen-decomposition, but this solution has serious scalability issues. In addition, how to impose regularizers on the sought canonical components was unclear – while structure-promoting regularizers are often desired in practice. We propose an algorithm that can easily handle datasets whose sample and feature dimensions are both large by exploiting data sparsity. The algorithm is also flexible in incorporating regularizers on the canonical components. Convergence properties of the proposed algorithm are carefully analyzed. Numerical experiments are presented to showcase its effectiveness.

***Index Terms*—** Generalized canonical correlation analysis, MAX-VAR, multi-view analysis

1. INTRODUCTION

In signal processing and data analytics, dimensionality reduction (DR) is usually the first step after signal and data acquisition. Principal component analysis (PCA) is arguably the most popular DR tool. However, PCA is designed to handle data that is acquired from a single feature domain. In modern data science, there are many cases where data have multiple representations in different domains – e.g., a word can be represented as an audio segment, an image, and some video frames. To integrate information from different feature spaces and extract informative low-dimensional representations, canonical correlation analysis (CCA) [1, 2] and generalized canonical correlation analysis (GCCA) [3, 4] are often applied. CCA is widely used in signal processing and data analytics; see [5–10].

Classical CCA considers two views (feature spaces) and formulates the corresponding DR problem as a generalized eigen-decomposition problem. On the other hand, GCCA considers more than two views, and various different formulations exist. Unlike CCA, most GCCA problems (e.g., the sum-of-correlations (SUM-COR) formulation) are NP-hard [11], and so approximations have been proposed to handle them. In the era of Big Data, both GCCA and plain CCA have serious scalability problems, since the computation involves inversion and square-root decomposition of cross-correlation matrices of the views – which is also referred to as the whitening process. Whitening destroys the sparsity of the views, which is often relied upon for dealing with big data, and also creates huge dense matrices that can hardly be stored and greatly increase the computational complexity of subsequent processing.

In recent years, scalability issues of CCA have drawn much attention, but most work focused on the two-view case [12–14]. In this work, we are interested in a popular formulation of GCCA, namely, the MAX-VAR GCCA. Unlike other GCCA formulations, MAX-VAR amounts to computing the leading eigenvectors of an aggregated and whitened correlation matrix of the views – and thus is optimally solvable. MAX-VAR GCCA has gained renewed interest in multilingual word embedding [15] and speech recognition [8], where it has demonstrated promising performance. However, MAX-VAR GCCA has the same scalability issues as the other formulations of (G)CCA. Another challenge of MAX-VAR GCCA is how to incorporate regularizers for promoting presumed or desired structure of the canonical components. Many regularizers are of interest; e.g., sparse canonical components can help discard outlying or irrelevant features, which is useful in gene studies [16–18]. Nonnegativity is of interest in video processing since nonnegative canonical components produce interpretable reduced-dimension data [19, 20].

To address the above challenges, we formulate structure-regularized MAX-VAR GCCA as a non-convex optimization problem and propose an alternating optimization (AO)-based algorithm to handle it. The algorithm alternates between a regularized least squares subproblem and a manifold-constrained non-convex subproblem. This way, the whitening matrices never need to be instantiated and the sparsity of the views is maintained – and thus the algorithm is highly scalable. Analogous to the proximal gradient, the algorithm can handle a variety of structure-promoting regularizers easily. We also carefully study the convergence properties of the proposed algorithm. We show that even when the two subproblems are inexactly solved, the algorithm converges to a critical point globally at a sublinear rate. When the classic MAX-VAR GCCA (without regularization) is considered, we further show that the algorithm in fact approaches a global minimum at a linear rate. Simulations show that the algorithm can easily scale up to views with $\sim 100,000$ samples and features, which is a substantial improvement from the classic solution that is only suitable for problem sizes of $\sim 1,000$.

2. BACKGROUND AND PROBLEM STATEMENT

The classic two-view CCA can be expressed as follows [1]:

$$\min_{\boldsymbol{Q}_1,\boldsymbol{Q}_2} \ \|\boldsymbol{X}_1\boldsymbol{Q}_1 - \boldsymbol{X}_2\boldsymbol{Q}_2\|_F^2 \tag{1a}$$

$$\text{s.t. } \boldsymbol{Q}_i^T\left(\boldsymbol{X}_i^T\boldsymbol{X}_i\right)\boldsymbol{Q}_i = \boldsymbol{I}, \quad i = 1, 2, \tag{1b}$$

where $\boldsymbol{X}_i \in \mathbb{R}^{L\times M_i}$ represents the ith view, $\boldsymbol{X}_i(\ell, :)$ is the high-dimensional data representation of entity (e.g., word) ℓ in view i, L and M_i denote the number of entities and the dimension of the ith

feature space, respectively, $\boldsymbol{Q}_i \in \mathbb{R}^{M_i \times K}$ contains the canonical components of the ith view that we aim at finding, and K is the dimension of the reduced-dimension views. Note that (1) essentially aims at maximizing the correlation of $\boldsymbol{X}_1\boldsymbol{Q}_1$ and $\boldsymbol{X}_2\boldsymbol{Q}_2$, which is the reason why the problem is called "correlation analysis". Problem (1) can be solved via the generalized eigen-decomposition, but this only applies to the two-view case. For dealing with $I \geq 2$ views, GCCA cost functions such as $\sum_{i=1}^{I-1}\sum_{j=i+1}^{I}\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{X}_j\boldsymbol{Q}_j\|_F^2$ (subject to $\boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_i\boldsymbol{Q}_i = \boldsymbol{I}$) are considered in the literature. Unlike the two-view case, such a pairwise matching criterion has been shown to be NP-hard [11]. Another formulation of GCCA is more tractable [3, 4, 8, 15, 21]:

$$\min_{\{\boldsymbol{Q}_i\}_{i=1}^I, \boldsymbol{G}^T\boldsymbol{G}=\boldsymbol{I}} \sum_{i=1}^{I} (1/2)\left\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}\right\|_F^2, \tag{2}$$

where $\boldsymbol{G} \in \mathbb{R}^{L\times K}$ is a common latent representation of the different views. Problem (2) also aims to find highly correlated reduced-dimension views but a "bridging variable" $\boldsymbol{G}$ is introduced for "coalescing" the multiple difficult constraints $\boldsymbol{Q}_i^T\boldsymbol{X}_i^T\boldsymbol{X}_i\boldsymbol{Q}_i = \boldsymbol{I}$ to a single constraint $\boldsymbol{G}^T\boldsymbol{G} = \boldsymbol{I}$. By doing so, the above problem admits a *conceptually* simple algebraic solution.

Problem (2) is referred to as the MAX-VAR formulation of GCCA in the literature [15]. To see the solution, let us first assume that $\boldsymbol{X}_i$ has full column rank and solve (2) with respect to (w.r.t.) $\boldsymbol{Q}_i$; i.e., $\boldsymbol{Q}_i = \boldsymbol{X}_i^\dagger\boldsymbol{G}$, where $\boldsymbol{X}_i^\dagger = (\boldsymbol{X}_i^T\boldsymbol{X}_i)^{-1}\boldsymbol{X}_i^T$. By substituting it back to (2), we see that an optimal solution $\boldsymbol{G}_{\rm opt}$ can be obtained via solving the following:

$$\boldsymbol{G}_{\rm opt} = \arg\max_{\boldsymbol{G}^T\boldsymbol{G}=\boldsymbol{I}} \ {\rm Tr}(\boldsymbol{G}^T\boldsymbol{M}\boldsymbol{G}), \tag{3}$$

where $\boldsymbol{M} = \sum_{i=1}^I \boldsymbol{X}_i\boldsymbol{X}_i^\dagger$. Then, an optimal solution is the first K principal eigenvectors of $\boldsymbol{M}$ [22].

Although the above solution to Problem (2) is seemingly easy, implementing it has two major challenges. First, there are serious scalability issues. Instantiating $\boldsymbol{M} = \sum_{i=1}^I \boldsymbol{X}_i(\boldsymbol{X}_i^T\boldsymbol{X}_i)^{-1}\boldsymbol{X}_i^T$ is not doable when L and the M_i's are large. The matrix $\boldsymbol{M}$ is an $L \times L$ *dense* matrix since $(\boldsymbol{X}_i^T\boldsymbol{X}_i)^{-1}$ is typically dense even when $\boldsymbol{X}_i$ is sparse. In applications like word embedding [15], L and M_i are both the vocabulary size of a language, which can easily exceed $100,000$. This means that the memory for simply instantiating $\boldsymbol{M}$ or $(\boldsymbol{X}_i^T\boldsymbol{X}_i)^{-1}$ can reach 75GB. Since sparsity is destroyed at the very beginning, the computational complexity of subsequent processing is also very high. Second, it is unclear how to incorporate regularization on $\boldsymbol{Q}_i$, since $\boldsymbol{Q}_i$ has been marginalized. Note that finding structured $\boldsymbol{Q}_i$ is well-motivated in practice. For example, when $\boldsymbol{X}_i$ has some outlying features (columns), a more appealing formulation may include a row sparsity-promoting regularization on $\boldsymbol{Q}_i$ so that those outlying columns in $\boldsymbol{X}_i$ can be discounted/downweighted when seeking $\boldsymbol{Q}_i$. Sparse (G)CCA is desired in many applications such as gene analytics and fMRI prediction [16–18, 23, 24]. Other structural constraints such as nonnegativity of $\boldsymbol{Q}_i$ is useful in data analytics for maintaining interpretability and enhancing performance; see [19, 20].

3. PROPOSED ALGORITHM

In this work, we consider a scalable and flexible algorithmic framework for handling MAX-VAR GCCA and its variants with structure-promoting regularizers on $\boldsymbol{Q}_i$. Specifically, we consider

$$\min_{\{\boldsymbol{Q}_i\}, \boldsymbol{G}^T\boldsymbol{G}=\boldsymbol{I}} \sum_{i=1}^{I} (1/2)\left\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}\right\|_F^2 + \sum_{i=1}^{I} g_i\left(\boldsymbol{Q}_i\right), \tag{4}$$

where $g_i(\cdot)$ is a regularizer that imposes a certain structure on $\boldsymbol{Q}_i$. In the literature, popular regularizers include $g_i(\boldsymbol{Q}_i) = \mu_i \cdot \|\boldsymbol{Q}_i\|_F^2$, $g_i(\boldsymbol{Q}_i) = \mu_i \cdot \|\boldsymbol{Q}_i\|_{2,1} = \mu_i \sum_{m=1}^M \|\boldsymbol{Q}_i(m,:)\|_2$, $g_i(\boldsymbol{Q}_i) = \mu_i \cdot \|\boldsymbol{Q}_i\|_{1,1} = \mu_i \sum_{m=1}^M \sum_{k=1}^K |\boldsymbol{Q}_i(m,k)|$, and $g_i(\boldsymbol{Q}_i) = \boldsymbol{1}_+(\boldsymbol{Q}_i)$; i.e., the indicator function of the nonnegative orthant, where $\mu_i \geq 0$ is a regularization parameter. We are particularly interested in $g_i(\boldsymbol{Q}_i) = \mu_i\|\boldsymbol{Q}_i\|_{2,1}$, since it has the ability of promoting rows of $\boldsymbol{Q}_i$ to be zero and thus can suppress the impact of the corresponding columns (features) in $\boldsymbol{X}_i$ – which effectively amounts to automatic joint feature selection together with GCCA. In this section, we propose an algorithm that can deal with the regularized and the original version of MAX-VAR GCCA under a unified framework.

3.1. Alternating Optimization

To deal with Problem (4), we build upon an alternating optimization (AO) framework. As we will see, this simple foundation enables us to design highly scalable algorithms in terms of both memory and computational cost, which also features great flexibility in incorporating regularization penalties.

Let us assume that after r iterations the current iterate is $(\{\boldsymbol{Q}_i^{(r)}\}, \boldsymbol{G}^{(r)})$. The subproblem w.r.t. $\boldsymbol{Q}_i$ is as follows:

$$\min_{\boldsymbol{Q}_i} \ (1/2)\left\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}^{(r)}\right\|_F^2 + g_i(\boldsymbol{Q}_i), \ \forall i. \tag{5}$$

When $\boldsymbol{X}_i$ is large and sparse, many efficient algorithms can be considered to solve the above – e.g., the alternating direction method of multipliers (ADMM) [25]. However, ADMM does not guarantee monotonic decrease of the objective value if the subproblem is not optimally solved. We wish to maintain monotonicity of the outer loop even when the subproblems are *inexactly* solved – note that inexact conditional updates are practically unavoidable when dealing with very large problems, for computational complexity considerations. This is an important difference when analyzing big sparse data. Hence, we propose to employ the proximal gradient (PG) method for handling Problem (5). Let us define $f_i(\boldsymbol{Q}_i, \boldsymbol{G}^{(r)}) = \frac{1}{2}\|\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{G}^{(r)}\|_F^2$ and $\nabla_{\boldsymbol{Q}_i} f_i(\boldsymbol{Q}_i, \boldsymbol{G}_i^{(r)}) = \boldsymbol{X}_i^T\boldsymbol{X}_i\boldsymbol{Q}_i - \boldsymbol{X}_i^T\boldsymbol{G}^{(r)}$. Then, by PG, we update $\boldsymbol{Q}_i$ by the following update rule:

$$\boldsymbol{Q}_i^{(r,t+1)} \leftarrow {\tt prox}_{g_i}\left(\boldsymbol{Q}_i^{(r,t)} - \alpha_i\nabla_{\boldsymbol{Q}_i} f_i\left(\boldsymbol{Q}_i^{(r,t)}, \boldsymbol{G}_i^{(r)}\right)\right), \tag{6}$$

where ${\tt prox}_{g_i}(\boldsymbol{Y}) = \arg\min_{\boldsymbol{X}} \ \|\boldsymbol{X} - \boldsymbol{Y}\|_2^2 + g_i(\boldsymbol{X})$, $\boldsymbol{Q}_i^{(r,t+1)}$ and $\boldsymbol{Q}_i^{(r,t)}$ denote $\boldsymbol{Q}_i$ at iteration $t+1$ and t when $\boldsymbol{G}^{(r)}$ is fixed, $t = 0, 1, \ldots, T-1$, and $\boldsymbol{Q}_i^{(r,0)} = \boldsymbol{Q}_i^{(r)}$ and $\boldsymbol{Q}_i^{(r,T)} = \boldsymbol{Q}_i^{(r+1)}$ under this notation. Note that we may choose a small T for efficiency. For many $g_i(\cdot)$'s including the aforementioned ones, the operator in (6) has closed-form or admits lightweight computation [26].

Next, we consider solving the subproblem w.r.t. $\boldsymbol{G}$ when fixing $\{\boldsymbol{Q}_i\}_{i=1}^I$. Instead of dealing with the original $\boldsymbol{G}$-subproblem, we propose to solve the following augmented form:

$$\min_{\boldsymbol{G}^T\boldsymbol{G}=\boldsymbol{I}} \ \sum_{i=1}^{I} 1/2\left\|\boldsymbol{X}_i\boldsymbol{Q}_i^{(r+1)} - \boldsymbol{G}\right\|_F^2 + \omega\|\boldsymbol{G} - \boldsymbol{G}^{(r)}\|_F^2, \tag{7}$$

where we define $\omega = {}^{(1-\gamma)I}/_{2\gamma}$ for $0 < \gamma \leq 1$. Note that when $\gamma = 1$, the above boils down to the original $\boldsymbol{G}$-subproblem. Adding the proximal term has the effect of controlling step size, which can lead to convergence rate guarantees as will be shown shortly. Expanding the above and dropping the constants, the solution amounts to the

following: Let $\boldsymbol{R} = \gamma \sum_{i=1}^{I} \boldsymbol{X}_i \boldsymbol{Q}_i^{(r+1)} / I + (1-\gamma)\boldsymbol{G}^{(r)}$. Then, an optimal solution of Problem (7) is

$$\boldsymbol{G}^{(r+1)} \leftarrow \boldsymbol{U}_R \boldsymbol{V}_R^T, \tag{8}$$

where $\boldsymbol{U}_R \boldsymbol{\Sigma}_R \boldsymbol{V}_R^T = \text{svd}(\boldsymbol{R}, 'econ')$, and $\text{svd}(\cdot, 'econ')$ denotes the economy-size SVD that produces $\boldsymbol{U}_R \in \mathbb{R}^{L \times K}$, $\boldsymbol{\Sigma}_R \in \mathbb{R}^{K \times K}$, and $\boldsymbol{V}_R^T \in \mathbb{R}^{K \times K}$. The above solution is based on the well-known Procrustes projection [27].

We call the proposed algorithm in Eqs (6) and (8) *alternating optimization-based MAX-VAR GCCA* (`AltMaxVar`). As one can see, the algorithm does not instantiate any large dense matrix during the procedure and thus is highly efficient in terms of memory. Also, the procedure does not destroy sparsity of the data, and thus the computational burden is light when the data is sparse – which is often the case in large-scale learning applications.

3.2. Computational and Memory Complexities

If the views $\boldsymbol{X}_i$ for $i = 1, \ldots, I$ are sparse, the PG updates are easy to compute. Specifically, when computing $\nabla_{\boldsymbol{Q}_i} f_i(\boldsymbol{Q}_i, \boldsymbol{G}_i)$, $\boldsymbol{X}_i \boldsymbol{Q}_i$ is calculated first, which has a complexity order of $\mathcal{O}(\text{nnz}(\boldsymbol{X}_i) \cdot K)$ flops, where $\text{nnz}(\cdot)$ counts the number of non-zeros. The next multiplication, i.e., $\boldsymbol{X}_i^T(\boldsymbol{X}_i \boldsymbol{Q}_i)$, has the same complexity order. The same applies to the operation of $\boldsymbol{X}_i^T \boldsymbol{G}$. For solving the $\boldsymbol{G}$-subproblem, since only an economy-size SVD of a very thin matrix (since $L \gg K$) is required, the step only costs $\mathcal{O}(LK^2)$ flops [22], which is linear in L.

In terms of memory, all the terms involved (i.e., $\boldsymbol{Q}_i$, $\boldsymbol{G}_i$, $\boldsymbol{X}_i \boldsymbol{Q}_i$, $\boldsymbol{X}_i^T \boldsymbol{X}_i \boldsymbol{Q}_i$ and $\boldsymbol{X}_i^T \boldsymbol{G}_i$) only require $\mathcal{O}(LK)$ memory or less, but the eigen-decomposition-based solution needs $\mathcal{O}(M_i^2)$ and $\mathcal{O}(L^2)$ memory to store $(\boldsymbol{X}_i^T \boldsymbol{X}_i)^{-1}$ and $\boldsymbol{M}$, respectively. Note that K is usually very small compared to L and M_i.

3.3. Convergence Properties

In this subsection, we present the results of convergence analysis of the proposed `AltMaxVar` algorithm. Due to space limitations, we must relegate all proofs to the forthcoming journal version [1]. Note that the algorithm alternates between a (possibly) non-smooth subproblem and a manifold-constrained subproblem, and the subproblems may or may not be solved to optimality. Existing convergence analyses for exact and inexact block coordinate descent such as those in [28–31] cannot be directly applied to analyze `AltMaxVar`, and thus its convergence properties are not obvious. We first establish convergence to a Karush-Kuhn-Tucker (KKT) point of Problem (4). A KKT point $(\boldsymbol{G}^*, \{\boldsymbol{Q}_i^*\}_i)$ satisfies the following first-order optimality conditions: $\boldsymbol{0} \in \nabla_{\boldsymbol{Q}_i} f_i(\boldsymbol{Q}_i^*, \boldsymbol{G}^*) + \partial_{\boldsymbol{Q}_i} g(\boldsymbol{Q}_i^*)$, $\forall i$ and $\boldsymbol{0} = \sum_{i=1}^{I} \nabla_{\boldsymbol{G}} f_i(\{\boldsymbol{Q}_i^*\}_i, \boldsymbol{G}^*) + \boldsymbol{G}\boldsymbol{\Lambda}^*$, $(\boldsymbol{G}^*)^T \boldsymbol{G}^* = \boldsymbol{I}$, where $\boldsymbol{\Lambda}$ is a Lagrangian multiplier associated with the constraint $\boldsymbol{G}^T \boldsymbol{G} = \boldsymbol{I}$, and $\partial_{\boldsymbol{Q}_i} g_i(\boldsymbol{Q}_i)$ denotes a set of subgradients of the (possibly) non-smooth function $g_i(\boldsymbol{Q}_i)$. We first show the following:

Proposition 1 *Assume that $\alpha_i \leq 1/L_i$ for all i, where $L_i = \lambda_{\max}(\boldsymbol{X}_i^T \boldsymbol{X}_i)$ is the largest eigenvalue of $\boldsymbol{X}_i^T \boldsymbol{X}_i$. Also assume that $g_i(\cdot)$ is a closed convex function, $T \geq 1$, and $\gamma \in (0, 1]$. Then, the following hold: (a) Every limit point of the solution sequence is a KKT point of Problem* (2)*. (b) If $\boldsymbol{X}_i$ and $\boldsymbol{Q}_i^{(0)}$ for $i = 1, \ldots, I$ are bounded and $\text{rank}(\boldsymbol{X}_i) = M_i$, then the whole solution sequence converges to the set $\mathcal{K}$ that consists of all the KKT points.*

Proposition 1 (a) characterizes the limit points of the solution sequence: Even if only one proximal gradient step is performed in each iteration r, every *convergent subsequence* of $\{\boldsymbol{G}^{(r)}, \{\boldsymbol{Q}_i^{(r)}\}_i\}_r$ attains a KKT point of Problem (4). Part (b) shows a stronger result regarding convergence of the *whole sequence*. The assumptions, on the other hand, are also more restrictive; i.e., $\text{rank}(\boldsymbol{X}_i) = M_i$.

It is also meaningful to estimate the number of iterations that is needed for the algorithm to reach a neighborhood of a KKT point. To this end, we show the following:

Theorem 1 *Assume that $\alpha_i < 1/L_i$, $0 < \gamma < 1$, and $T \geq 1$. Let $\delta > 0$ and J be the number of iterations needed so that $Z^{(r+1)} \leq \delta$ holds for the first time, where*

$$Z^{(r+1)} = \sum_{t=0}^{T-1} \sum_{i=1}^{I} \left\| \tilde{\nabla}_{\boldsymbol{Q}_i} F_i(\boldsymbol{Q}_i^{(r,t)}, \boldsymbol{G}^{(r)}) \right\|_F^2 + \left\| \boldsymbol{G}^{(r)} - \textstyle\sum_{i=1}^{I} \boldsymbol{X}_i \boldsymbol{Q}_i^{(r+1)} / I + \boldsymbol{G}^{(r+1)} \boldsymbol{\Lambda}^{(r+1)} \right\|_F^2,$$

in which $\tilde{\nabla}_{\boldsymbol{Q}_i} F_i(\boldsymbol{Q}_i^{(r,t)}, \boldsymbol{G}^{(r)}) = \frac{1}{\alpha_i}(\boldsymbol{Q}_i^{(r+1,t)} - \text{prox}_{g_i}(\boldsymbol{Q}_i^{(r,t)} - \alpha_i \nabla_{\boldsymbol{Q}_i} f(\boldsymbol{Q}_i^{(r,t)}, \boldsymbol{G}^{(r)})))$ is the proximal gradient at $\boldsymbol{Q}_i^{(r,t)}$ w.r.t. $\boldsymbol{Q}_i$, and $\boldsymbol{\Lambda}^{(r+1)}$ is the Lagrangian multiplier associated with $\boldsymbol{G}^{(r+1)}$. Then, there exists a constant $v \geq 0$ such that $\delta \leq v/J-1$.

In Theorem 1, the Z-function serves as a measure of the optimality gap between the current iterate and a KKT point since one can show that $Z^{(r+1)} \to 0$ implies that a KKT point is attained. By Theorem 1, `AltMaxVar` reduces the (measure of the) optimality gap to $\mathcal{O}(1/r)$ after r iterations – at least a sublinear rate is guaranteed. One subtle point worth mentioning is that the analysis in Theorem 1 holds when $\gamma < 1$, which corresponds to the case where $\omega > 0$ in the $\boldsymbol{G}$-subproblem and the solution is controlled to be not far away from $\boldsymbol{G}^{(r)}$. This reflects an interesting fact in AO – when the subproblems are handled in a conservative way using a controlled step size, a certain convergence rate property may be guaranteed.

Note that when $g_i(\cdot) = \mu_i \|\cdot\|_F^2$ for $\mu_i \geq 0$, this case is *optimally solvable* (i.e., the solution is the K leading eigenvectors of $\boldsymbol{M} = \sum_{i=1}^{I} \boldsymbol{X}_i (\boldsymbol{X}_i^T \boldsymbol{X}_i + \mu_i \boldsymbol{I})^{-1} \boldsymbol{X}_i^T$ for $\mu_i \geq 0$ [15]). Therefore, it is natural to ask if `AltMaxVar` loses optimality under such cases by gaining scalability? To address this question, we denote $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ as the K principal eigenvectors of $\boldsymbol{M}$ and the eigenvectors spanning its orthogonal complement, respectively. Recall that our goal is to find $\boldsymbol{G}$ such that $\mathcal{R}(\boldsymbol{G}) = \mathcal{R}(\boldsymbol{U}_1)$. We adopt the definition of subspace distance in [22], i.e., $\text{dist}(\mathcal{R}(\boldsymbol{G}^{(r)}), \mathcal{R}(\boldsymbol{U}_1)) = \|\boldsymbol{U}_2^T \boldsymbol{G}^{(r)}\|_2$, where $\|\boldsymbol{X}\|_2$ denotes the matrix 2-norm, and show the following:

Theorem 2 *Denote the eigenvalues of $\boldsymbol{M} \in \mathbb{R}^{L \times L}$ by $\lambda_1, \ldots, \lambda_L$ in descending order. Consider $g_i(\cdot) = \mu_i \|\cdot\|_F^2$ for $\mu_i \geq 0$ and let $\gamma = 1$. Assume that $\text{rank}(\boldsymbol{X}_i) = M_i$, $\lambda_K > \lambda_{K+1}$, and $\mathcal{R}(\boldsymbol{G}^{(0)})$ is not orthogonal to any component in $\mathcal{R}(\boldsymbol{U}_1)$; i.e., $\cos(\theta) = \min_{\boldsymbol{u} \in \mathcal{R}(\boldsymbol{U}_1), \boldsymbol{v} \in \mathcal{R}(\boldsymbol{G}^{(0)})} \frac{|\boldsymbol{u}^T \boldsymbol{v}|}{(\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2)} > 0$. Also assume that each subproblem in* (5) *is solved to accuracy ϵ; i.e., $\|\boldsymbol{Q}_i^{(r+1)} - \tilde{\boldsymbol{Q}}_i^{(r+1)}\|_2 \leq \epsilon$, where $\tilde{\boldsymbol{Q}}_i^{(r+1)} = (\boldsymbol{X}_i^T \boldsymbol{X}_i + \mu_i \boldsymbol{I})^{-1} \boldsymbol{X}_i \boldsymbol{G}^{(r)}$. Then, after r iterations,*

$$\text{dist}\left(\mathcal{R}(\boldsymbol{G}^{(r)}), \mathcal{R}(\boldsymbol{U}_1)\right) \leq \tan(\theta) \left(\lambda_{K+1}/\lambda_K\right)^r + C$$

holds, where $C = \mathcal{O}\left(\sum_{i=1}^{I} \lambda_{\max}(\boldsymbol{X}_i)\epsilon\right)$ is a constant.

Theorem 2 ensures that if a T suffices for the $\boldsymbol{Q}$-subproblem to obtain a good enough approximation to its optimal solution, the algorithm converges *linearly* to a *global optimal* solution up to some accuracy loss. In our simulations, we observe that using $T = 1$ already gives very satisfactory results (as will be shown in the next section), which leads to computationally very cheap updates.

[1] A longer version of the paper with more detailed proofs is available at `http://arxiv.org/abs/1605.09459`.

4. SIMULATIONS AND CONCLUSIONS

We generate the views by $\boldsymbol{X}_i = \boldsymbol{Z}\boldsymbol{A}_i + \sigma\boldsymbol{N}_i$ where $\boldsymbol{Z} \in \mathbb{R}^{L\times N}$ is common to all views, $\boldsymbol{A}_i \in \mathbb{R}^{N\times M}$ is a "mixing matrix" whose effect is supposed to be suppressed by $\boldsymbol{Q}_i$, $\boldsymbol{N}_i \in \mathbb{R}^{L\times M}$ is noise, and $\sigma \geq 0$. $\boldsymbol{Z}$, $\boldsymbol{A}_i$, and $\boldsymbol{N}_i$ are large sparse matrices and the non-zero elements follow the zero-mean unit-variance i.i.d. Gaussian distribution. $\boldsymbol{X}_i$ is sparse and its density level ρ_i is definied as $\rho_i = {\rm nnz}(\boldsymbol{X}_i)/LM$. In the simulations, we let $\rho = \rho_i$ for all i. We use the eigen-decomposition based solution of MAX-VAR GCCA as a benchmark when applicable. Another algorithm called multiview latent semantic analysis (MVLSA) is also employed as a baseline [15]. MVLSA truncates the rank of the views using PCA as pre-processing and then applies the eigen-decomposition based solution.

In Fig. 1, we show the runtime performance of the algorithms for various sizes of the views, where density of the views is controlled so that $\rho \approx 10^{-3}$. The regularization $g_i(\cdot) = 0.1\|\cdot\|_F^2$ is employed by all algorithms. We let $M = L \times 0.8$, $M = N$ and change M from $5,000$ to $50,000$. To run MVLSA, we truncate the ranks of views to $P = 100$, $P = 500$ and $P = 1,000$, respectively. We use MVLSA with $P = 100$ to initialize AltMaxVar and let $T = 1$ and $\gamma = 1$. We stop the proposed algorithm when the absolute change of the objective value is smaller than 10^{-4}. Ten random trials are used to obtain the results. One can see that the eigen-decomposition based algorithm does not scale well since the matrix $(\boldsymbol{X}_i^T\boldsymbol{X}_i + \mu_i\boldsymbol{I})^{-1}$ is dense. In particular, the algorithm exhausts the memory quota (32GB RAM) when $M = 30,000$. MVLSA with $P = 100$ and the proposed algorithm both scale very well from $M = 5,000$ to $M = 50,000$: When $M = 20,000$, brute-force eigen-decomposition takes almost 80 minutes, whereas MVLSA ($P = 100$) and AltMaxVar both use less than 2 minutes. Note that the runtime of the proposed algorithm already includes the runtime of the initialization time by MVLSA with $P = 100$, and thus the runtime curve of AltMaxVar is slightly higher than that of MVLSA ($P = 100$) in Fig. 1. Another observation is that, although MVLSA exhibits good runtime performance when using $P = 100$, its runtime under $P = 500$ and $P = 1,000$ is not very appealing. The corresponding cost values can be seen in Table 1. The eigen-decomposition based method gives the lowest cost values when applicable, as it is an optimal solution. The proposed algorithm gives favorable cost values that are close to the optimal ones, even when only one iteration of the $\boldsymbol{Q}$-subproblem is implemented for every fixed $\boldsymbol{G}^{(r)}$ – this result supports our analysis in Theorem 2. Increasing P helps improve MVLSA. However, even when $P = 1,000$, the cost value given by MVLSA is still higher than that of AltMaxVar, and MVLSA using $P = 1,000$ is much slower than AltMaxVar.

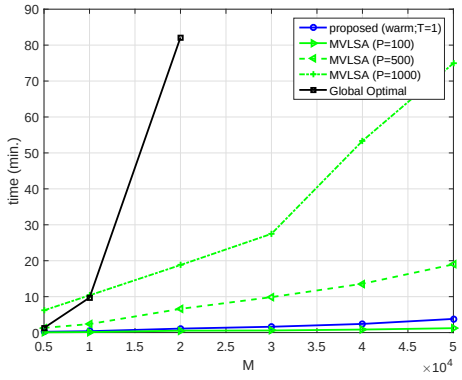

Fig. 1: Runtime of the algorithms under various problem sizes.

Table 1: Cost values of the algorithms under different problem sizes. $L = M/0.8$, $\rho = 10^{-3}$, $\sigma = 0.1$. † means "out of memory".

Algorithm	M					
	5,000	10,000	20,000	30,000	40,000	50,000
Global Opt	0.053	0.033	0.021	†	†	†
MVLSA ($P = 100$)	2.164	3.527	5.065	5.893	6.475	7.058
MVLSA ($P = 500$)	0.280	0.717	1.766	2.582	3.407	3.996
MVLSA ($P = 1,000$)	0.125	0.287	0.854	1.406	2.012	2.513
Proposed	**0.092**	**0.061**	**0.049**	**0.043**	**0.038**	**0.039**

Table 2 presents the simulation results of a large-scale case in the presence of outlying features. Here, we fix $L = 100,000$ and $M = 80,000$ and change the density level ρ. We add $30,000$ outlying features to each view and every outlying feature is a random sparse vector whose non-zero elements follow the zero-mean i.i.d. unit-variance Gaussian distribution. We also scale the outlying features so that the average energy of the clean and outlying features are identical. The other settings follow those in the last simulation. In this case, the optimal solution to Problem (4) is unknown. Therefore, we evaluate the performance by observing ${\rm metric}_1 = {}^1\!/_I \sum_{i=1}^I \|\boldsymbol{X}_i(:,\mathcal{S}_i^c)\hat{\boldsymbol{Q}}_i(\mathcal{S}_i^c,:) - \hat{\boldsymbol{G}}\|_F^2$, and ${\rm metric}_2 = {}^1\!/_I \sum_{i=1}^I \|\boldsymbol{X}_i(:,\mathcal{S}_i)\hat{\boldsymbol{Q}}_i(\mathcal{S}_i,:)\|_F^2$, where $\mathcal{S}_i^c$ and $\mathcal{S}_i$ denote the index sets of clean and outlying features of view i, respectively – i.e., $\boldsymbol{X}_i(:,\mathcal{S}_i^c) = \boldsymbol{Z}_i\boldsymbol{A}_i$ and $\boldsymbol{X}_i(:,\mathcal{S}_i) = \boldsymbol{O}_i$ if noise is absent. ${\rm metric}_1$ measures the performance of matching $\hat{\boldsymbol{G}}$ with the relevant part of the views, while ${\rm metric}_2$ measures the performance of suppressing the irrelevant part. We desire low values of ${\rm metric}_1$ and ${\rm metric}_2$ simultaneously. We use $g_i(\cdot) = \mu_i\|\cdot\|_{2,1}$ for AltMaxVar to discard the outlying features. One can see from Table 2 that the proposed algorithm with $\mu_i = 0.05$ gives the most balanced result – both evaluation metrics are at fairly low levels. Using $\mu_i = 0.5$ suppresses $\boldsymbol{Q}_i(\mathcal{S},:)$ even better, but using such a relatively large μ_i degrades the fitting metric. In terms of runtime, one can see that the proposed algorithm operates within the same order of magnitude as MVLSA with $P = 100$. Since AltMaxVar works with the intact views of size $L \times M$ while MVLSA works with significantly reduced-dimension data, such runtime performance of AltMaxVar is very satisfactory.

Table 2: Evaluation in the presence of outlying features. $L = 100,000$, $M = 80,000$, $|\mathcal{S}| = 30,000$, $\sigma = 1$, $I = 3$.

Algorithm	measure	ρ (density of views)			
		10^{-5}	5×10^{-4}	10^{-4}	10^{-3}
MVLSA ($P = 100$)	metric1	16.843	13.877	17.159	16.912
	metric2	**0.003**	**0.010**	0.009	0.003
	time (min)	**0.913**	**1.019**	**1.252**	**3.983**
Proposed ($\mu = .05$)	metric1	**0.478**	**0.610**	**0.565**	**0.775**
	metric2	0.018	0.134	0.034	0.003
	time (min)	3.798	5.425	5.765	24.182
Proposed ($\mu = .1$)	metric1	0.942	1.054	0.941	1.265
	metric2	0.006	0.054	**0.004**	**0.000**
	time (min)	2.182	3.791	4.510	16.378
Proposed ($\mu = .5$)	metric1	1.592	1.497	1.306	1.538
	metric2	**0.003**	0.021	**0.000**	**0.000**
	time (min)	1.735	2.714	3.723	13.447

To conclude, we have considered large-scale MAX-VAR GCCA with structure-promoting regularization and designed a memory-efficient and computationally lightweight algorithm that easily incorporates various common regularization penalties. Our analysis shows that the algorithm converges to a KKT point for a variety of regularizations at a sublinear rate. When the classic MAX-VAR GCCA is considered, the algorithm approaches a global optimal solution at a linear rate. Simulations demonstrated the good scalability and the effectiveness of the proposed algorithm.

5. REFERENCES

[1] 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.

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

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

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

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

[6] A. Bertrand and M. Moonen, "Distributed canonical correlation analysis in wireless sensor networks with application to distributed blind source separation," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4800–4813, 2015.

[7] A. Dogandzic and A. Nehorai, "Finite-length mimo equalization using canonical correlation analysis," *IEEE Trans. Signal Process.*, vol. 50, no. 4, pp. 984–989, 2002.

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

[9] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, "Acoustic segment modeling with spectral clustering methods," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 2, pp. 264–277, 2015.

[10] P. M. Djurić and Y. Wang, "Distributed bayesian learning in multiagent systems: Improving our understanding of its capabilities and limitations," *IEEE Signal Process. Mag.*, vol. 29, no. 2, pp. 65–76, March 2012.

[11] 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.

[12] Z. Ma, Y. Lu, and D. Foster, "Finding linear structure in large datasets with scalable canonical correlation analysis," *arXiv preprint arXiv:1506.08170*, 2015.

[13] L. Sun, S. Ji, and J. Ye, "Canonical correlation analysis for multilabel classification: A least-squares formulation, extensions, and analysis," *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 33, no. 1, pp. 194–200, 2011.

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

[15] P. Rastogi, B. Van Durme, and R. Arora, "Multiview LSA: Representation learning via generalized cca," in *Proc. NAACL*, 2015.

[16] D. M. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, p. kxp008, 2009.

[17] X. Chen, H. Liu, and J. G. Carbonell, "Structured sparse canonical correlation analysis," in *International Conference on Artificial Intelligence and Statistics*, 2012, pp. 199–207.

[18] 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.

[19] C. Sigg, B. Fischer, B. Ommer, V. Roth, and J. Buhmann, "Nonnegative cca for audiovisual source separation," in *2007 IEEE Workshop on Machine Learning for Signal Processing*. IEEE, 2007, pp. 253–258.

[20] B. Fischer, V. Roth, and J. M. Buhmann, "Time-series alignment by non-negative multiple generalized canonical correlation analysis," *BMC bioinformatics*, vol. 8, no. Suppl 10, p. S4, 2007.

[21] M. Van De Velden and T. H. A. Bijmolt, "Generalized canonical correlation analysis of matrices with missing rows: a simulation study," *Psychometrika*, vol. 71, no. 2, pp. 323–331, 2006.

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

[23] I. Rustandi, M. A. Just, and T. Mitchell, "Integrating multiple-study multiple-subject fmri datasets using canonical correlation analysis," in *Proceedings of the MICCAI 2009 Workshop: Statistical modeling and detection issues in intra-and inter-subject functional MRI data analysis*, 2009.

[24] T. M. Mitchell, S. V. Shinkareva, A. Carlson, K.-M. Chang, V. L. Malave, R. A. Mason, and M. A. Just, "Predicting human brain activity associated with the meanings of nouns," *science*, vol. 320, no. 5880, pp. 1191–1195, 2008.

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

[27] P. Schönemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[28] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

[29] 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.

[30] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM Journal on imaging sciences*, vol. 6, no. 3, pp. 1758–1789, 2013.

[31] ——, "A globally convergent algorithm for nonconvex optimization based on block coordinate update," *arXiv preprint arXiv:1410.1386*, 2014.