Common Component Analysis for Multiple Covariance Matrices

Huahua Wang
Computer Science & Engg
Univ of Minnesota, Twin Cities
huwang@cs.umn.edu

Arindam Banerjee
Computer Science & Engg
Univ of Minnesota, Twin Cities
banerjee@cs.umn.edu

Daniel Boley
Computer Science & Engg
Univ of Minnesota, Twin Cities
boley@cs.umn.edu

ABSTRACT

We consider the problem of finding a suitable common low dimensional subspace for accurately representing a given set of covariance matrices. With one covariance matrix, this is principal component analysis (PCA). For multiple covariance matrices, we term the problem *Common Component Analysis* (CCA). While CCA can be posed as a tensor decomposition problem, standard approaches to tensor decompositions have two critical issues: (i) tensor decomposition methods are iterative and rely on the initialization; (ii) for a given level of approximation error, it is difficult to choose a suitable low dimensionality. In this paper, we present a detailed analysis of CCA that yields an effective initialization and iterative algorithms for the problem. The proposed methodology has provable approximation guarantees w.r.t. the global maximum and also allows one to choose the dimensionality for a given level of approximation error. We also establish conditions under which the methodology will achieve the global maximum. We illustrate the effectiveness of the proposed method through extensive experiments on synthetic data as well as on two real stock market datasets, where major financial events can be visualized in low dimensions.

Categories and Subject Descriptors: H.2.8 **[Database Management]:** Database applications – Data mining
General Terms: Algorithms
Keywords: PCA, Dimensionality Reduction, Tensor Decompositions, PARAFAC, TUCKER

1. INTRODUCTION

In recent years, simultaneous analysis of multiple high dimensional covariance matrices is becoming increasingly important in diverse application domains ranging from finance to climate and environmental sciences [30, 31, 32, 11, 34]. The traditional approach for finding an accurate low-dimensional approximation to a high-dimensional covariance matrix is principal component analysis (PCA) [14, 4]. In particular, PCA finds an orthogonal projection of a single covariance matrix to a low-dimensional space while preserving as much of the "energy" or variance as possible. The problem can be solved by performing the eigenvalue decomposition (EVD) on the single covariance matrix under consideration.

Given multiple covariance matrices, we consider the problem of finding a suitable common low-dimensional subspace for accurately representing all the covariance matrices. We term the problem *Common Component Analysis* (CCA). PCA is not suitable for finding such a subspace for multiple covariance matrices, particularly if the covariance matrices span different subspaces. Examples include stock market data where financial shocks and volatility arise from different sources, and yield stock return covariance matrices in different subspaces. The low-dimensional covariance representation of the high-dimensional covariance matrices can take two possible forms: diagonal or full. Existing models where diagonal low rank matrices are considered, such as PARAFAC/ CANDECOMP [16, 17, 24, 22] and common principal components (CPC) [12, 13], do not allow interactions among low-dimensional components and essentially assume that underlying factors are uncorrelated. Moreover, multiple matrices can be simultaneously diagonalized with an orthogonal transformation if and only if they commute [19], which need not be true in general. Consequently, in this paper, we consider the case where the low-dimensional covariance matrices are full matrices. Such decompositions have been widely studied under different names, such as the Tucker2 model [35, 16, 24, 25, 22], tensor PCA [6], 2DSVD [9], GLRAM [37], and tensor decompositions [21, 22, 26, 33]. Variance correlation [11] and Cholesky decomposition [3, 31] have also been used to simultaneously model multiple covariance matrices in low dimensions.

While CCA can be posed as a tensor decomposition problem, unlike PCA, standard approaches to tensor decompositions have two critical issues: (i) tensor decomposition methods are iterative and rely on the initialization; (ii) for a given level of approximation error, efficiently choosing a suitable low dimensionality is difficult in general. In this paper, we present a detailed analysis that alleviates the two issues in the context of CCA. We start by showing that our problem is equivalent to maximizing (not minimizing) a convex function over a compact but non-convex set. As a result, finding the global maximum is difficult in general. With an analysis using a simpler variant of CCA, we derive lower and upper bounds for the CCA objective for any orthonormal matrix. The bounds naturally lead to corresponding lower and upper bounds for the global maximum of CCA. We also give sufficient conditions under which the global maximum will be achieved. In [9], similar bounds were established for a local maximum of a related problem, but the closeness of the bounds w.r.t. the global maximum was not explicitly investigated. Using our bounds, an effective initialization is proposed. It has been observed that a similar initialization often leads to the global maximum [37, 9], particularly for rank-1 approximations [27, 20]. Our analysis shows that instead of starting with a given low dimensionality, one can start with an approximation error bound, and appropriately choose a sufficient dimension-

KDD'11, August 21–24, 2011, San Diego, California, USA.

ality for CCA satisfying the given error bound. We present two algorithms to iteratively improve the objective for CCA starting from the proposed initialization. One algorithm adopts an existing idea from tensor decompositions [24, 25, 6, 9, 37]. In each iteration, the update in the standard tensor decomposition algorithm requires computing the EVD of an $n \times n$ matrix, where n is the dimensionality of the observed high-dimensional covariance matrices. We also propose a novel algorithm based on an auxiliary function [28, 29]. In each iteration, the update in the novel algorithm only requires performing the singular value decomposition (SVD) of an $r \times n$ matrix, where r is the dimensionality of the latent low-dimensional covariance matrices. When $r \ll n$, the auxiliary-function-based algorithm is substantially more efficient than the standard tensor decomposition algorithm.

The remainder of this paper is organized as follows. We formulate the common component analysis (CCA) problem in Section 2. In Section 3, we analyze the problem, establish lower and upper bounds for the global maximum, introduce the initialization and its optimality properties, establish sufficient conditions under which the global maximum will be achieved, and also discuss the connections to related work. In Section 4, we present two algorithms for CCA given a suitable initialization, which can work with a given dimensionality or a given approximation error bound. We report experimental results on synthetic data as well as two stock market datasets to illustrate the performance of the proposed ideas in Section 5, and conclude in Section 6.

Notation: Matrices are denoted by uppercase bold letters, e.g., $\mathbf{X}, \mathbf{U}$, etc. The diagonal entries in a diagonal matrix are generally assumed to be in non-decreasing order, especially if arising from the EVD or SVD. $\mathbb{I}_r$ (with r an integer) denotes a $r \times r$ identity matrix.

2. PROBLEM FORMULATION

Assume we have a set of high-dimensional covariance matrices $\mathbf{X}_t \in \mathbb{R}^{n \times n}, 1 \leq t \leq T$. The key hypothesis driving our analysis is that the high-dimensional covariance matrices are indeed linearly transformed versions of a set of low-dimensional covariance matrices $\mathbf{Y}_t \in \mathbb{R}^{r \times r}, 1 \leq t \leq T$. While the linear transformation $\mathbf{U} \in \mathbb{R}^{n \times r}$ as well as the low-dimensional covariance matrices $\mathbf{Y}_t, 1 \leq t \leq T$, are unknown, each $\mathbf{X}_t$ is assumed to be well approximated by $\mathbf{U}\mathbf{Y}_t\mathbf{U}^T$. In particular,

$$\mathbf{X}_t = \mathbf{U}\mathbf{Y}_t\mathbf{U}^T + \mathbf{E}_t \,, \tag{1}$$

where $\mathbf{E}_t$ is the residual matrix. Without loss of generality, $\mathbf{U}$ is assumed to be orthonormal, i.e., $\mathbf{U}^T\mathbf{U} = \mathbb{I}_r$. The goal is to find $\mathbf{U}$ and $\mathbf{Y}_t, 1 \leq t \leq T$ such that the sum of squares of the Frobenius norms of all the residual matrices is minimized. The problem can be formally stated as follows:

$$\min_{\substack{\mathbf{U},\mathbf{Y}_t \\ \mathbf{U}^T\mathbf{U}=\mathbb{I}_r}} \sum_{t=1}^{T} \|\mathbf{X}_t - \mathbf{U}\mathbf{Y}_t\mathbf{U}^T\|_F^2 \,. \tag{2}$$

Since $\mathbf{U}$ determines a common subspace for all the covariance matrices, we call the above formulation *Common Component Analysis* (CCA). There are several appealing properties for the choices of orthonormal matrix $\mathbf{U}$ and non-diagonal matrix $\mathbf{Y}_t$. First, since $\mathbf{U}$ is orthonormal, the formulation allows one to visualize covariance matrices in the low-dimensional subspaces (see Figure 4 in Section 5.2). With non-orthonormal $\mathbf{U}$, such low-dimensional covariance matrices can be difficult to interpret. Second, it turns out that the choices can reduce CCA to a maximization problem over $\mathbf{U}$ (see Lemma 1) by dropping out $\mathbf{Y}_t$, thus facilitating a theoretical analysis. Further, the full matrix $\mathbf{Y}_t$ allows interactions among components and leads to substantially lower approximation errors (see Section 5.1 and 5.2).

We make a few observations before continuing with our analysis. If there is only one covariance matrix $\mathbf{X}_1$ under consideration, then the model reduces to standard PCA. If $\mathbf{X}_t$ is not a covariance matrix, i.e., $\mathbf{X}_t \in \mathbb{R}^{m \times n}$, it is modeled as $\mathbf{X}_t = \mathbf{U}\mathbf{Y}_t\mathbf{V}^T + \mathbf{E}_t$, where $\mathbf{U} \in \mathbb{R}^{m \times r}, \mathbf{Y}_t \in \mathbb{R}^{r \times s}, \mathbf{V} \in \mathbb{R}^{s \times n}$, and the existing literature on tensor decompositions is relevant [24, 26, 22, 21, 6, 9, 37, 16, 17, 35]. Assuming $r = s$ and restricting $\mathbf{Y}_t$ to be diagonal leads to the PARAFAC/CANDECOMP models [24, 22]. When such restrictions are not imposed, one gets the Tucker2 model [24, 22]. Iterative algorithms and data mining applications of such decompositions have been studied in the literature [24, 22, 23, 9, 37]. Unlike most existing settings, in our model each $\mathbf{X}_t$ is a positive semi-definite matrix, and $\mathbf{Y}_t$ is also positive semi-definite. In particular, CCA is different from CPC [12, 13]. While CPC aims to simultaneously diagonalize a set of strictly positive definite covariance matrices using a maximum likelihood approach [13], CCA is discussed in a least squares setting. We discuss technical relationships of our analysis to existing models in Section 3.5.

We start the analysis with the following two results. Space does not permit us to include the proofs, which can be found in [36].

Lemma 1 *The optimum $\mathbf{Y}_t$ in (2) satisfies $\mathbf{Y}_t = \mathbf{U}^T\mathbf{X}_t\mathbf{U}$. Further, the optimal $\mathbf{U}$ in (2) is the solution to the following problem:*

$$\max_{\mathbf{U}^T\mathbf{U}=\mathbb{I}_r} f(\mathbf{U}) = \max_{\mathbf{U}^T\mathbf{U}=\mathbb{I}_r} \operatorname{Tr}(\mathbf{U}^T M(\mathbf{U})\mathbf{U}) \,, \tag{3}$$

where

$$M(\mathbf{U}) = \sum_{t=1}^{T} \mathbf{X}_t\mathbf{U}\mathbf{U}^T\mathbf{X}_t \,. \tag{4}$$

Lemma 2 *For $\mathbf{U} \in \mathbb{R}^{n \times r}$ (not necessarily orthonormal), $f(\mathbf{U})$ is a convex function.*

Unfortunately, the fact that $f(\mathbf{U})$ is convex does not help us because we are *maximizing* $f(\mathbf{U})$ in (3) instead of minimizing it. Further, the constraint set $\mathbf{U}^T\mathbf{U} = \mathbb{I}_r$ is not convex. As a result, the problem in (3) is not convex. In fact, the problem is one of maximizing a convex function over a bounded non-convex feasible set. As a result, there may be several local maxima. If starting from an initial guess, the standard approaches to tensor decompositions will likely get stuck in a local maximum. Furthermore, it is difficult to characterize the proximity of such solutions in terms of the function value achieved with respect to the global maximum. In the next two sections, we develop a simple way to initialize $\mathbf{U}$ along with algorithms for iterative updates with guarantees relative to the global maximum.

3. ANALYSIS

In this section, we analyze CCA in terms of a simpler model called common component analysis 1 (CCA1). We show that CCA1 is a PCA-style problem and can be solved using the EVD. More importantly, the solution to CCA1 leads to lower and upper bounds on the global maximum of CCA and suggests a good initialization for any iterative algorithm for CCA. We also show how to choose a suitable dimensionality sufficient to satisfy a given approximation error bound. In addition, sufficient conditions for a global maximum and the connections to related work are considered.

Table 1: CCA and CCA1

CCA	CCA1
$\mathbf{X}_t = \mathbf{U}\mathbf{Y}_t\mathbf{U} + \mathbf{E}_t$	$\mathbf{X}_t = \mathbf{U}\mathbf{Z}_t + \mathbf{E}_t$
$M(\mathbf{U}) = \sum_t \mathbf{X}_t\mathbf{U}\mathbf{U}^T\mathbf{X}_t$	$M(\mathbb{I}_\mathbf{n}) = \sum_t \mathbf{X}_t^2$
$f(\mathbf{U}) = \mathrm{Tr}(\mathbf{U}^T M(\mathbf{U})\mathbf{U})$	$f_1(\mathbf{U}) = \mathrm{Tr}(\mathbf{U}^T M(\mathbb{I}_\mathbf{n})\mathbf{U})$

3.1 A Simpler Model: CCA1

Instead of the original problem in (2), we consider a simpler decomposition given by

$$\mathbf{X}_t = \mathbf{U}\mathbf{Z}_t + \mathbf{E}_t \ , \qquad (5)$$

where $\mathbf{U} \in \mathbb{R}^{n\times r}$ and $\mathbf{Z}_t \in \mathbb{R}^{r\times n}$. Assuming the residual norms to be small, the problem of finding $\mathbf{U}, \mathbf{Z}_t$ can be posed as follows:

$$\min_{\substack{\mathbf{U},\mathbf{Z}_t \\ \mathbf{U}^T\mathbf{U}=\mathbb{I}_r}} \sum_{t=1}^{T} \|\mathbf{X}_t - \mathbf{U}\mathbf{Z}_t\|_F^2 \ . \qquad (6)$$

We call the above problem CCA1 since it only considers a one-sided projection compared to the two-sided projection in CCA. Similar to CCA, the simplified problem CCA1 allows an alternative characterization as follows:

Lemma 3 *The optimal $\mathbf{Z}_t$ in (6) satisfies $\mathbf{Z}_t = \mathbf{U}^T\mathbf{X}_t$. Furthermore, the optimal $\mathbf{U}$ in (5) is the solution to the following problem:*

$$\max_{\mathbf{U}^T\mathbf{U}=\mathbb{I}_r} f_1(\mathbf{U}) = \max_{\mathbf{U}^T\mathbf{U}=\mathbb{I}_r} \mathrm{Tr}(\mathbf{U}^T M(\mathbb{I}_n)\mathbf{U}) \ , \qquad (7)$$

where $\mathbb{I}_n$ is an identity matrix of size n and

$$M(\mathbb{I}_n) = \sum_{t=1}^{T} \mathbf{X}_t^2 \ . \qquad (8)$$

Note that CCA1 in (7) is a PCA problem on $M(\mathbb{I}_n)$, which can be solved using the EVD. Table 1 shows a relative comparison between CCA and CCA1.

3.2 Lower and Upper Bounds

The solution of CCA1 helps significantly in characterizing the solution to CCA. We focus on developing lower and upper bounds on the global maximum of CCA based on the solution of CCA1. Since CCA1 is essentially the PCA problem over $M(\mathbb{I}_n) = \sum_t \mathbf{X}_t^2$, if $\mathbf{U}_0$ denotes the top r eigenvectors of $M(\mathbb{I}_n) = \sum_{t=1}^{T} \mathbf{X}_t^2$, then $\mathbf{U}_0$ is the solution to (7). Let $f_1^{\max} = f_1(\mathbf{U}_0)$ be the maximum value of $f_1(\mathbf{U})$, and let $M_T = \mathrm{Tr}(M(\mathbb{I}_n)) = \mathrm{Tr}\left(\sum_t \mathbf{X}_t^2\right)$. With this notation, we have the following result:

Theorem 1 *Let $M_T = \mathrm{Tr}(\sum_t \mathbf{X}_t^2)$. Then, with $f_1(\mathbf{U})$ and $f(\mathbf{U})$ denoting the objective functions for CCA1 and CCA respectively as in (7) and (3), for any $\mathbf{U}$ with $\mathbf{U}^T\mathbf{U} = \mathbb{I}_r$, we have*

$$\frac{f_1^2(\mathbf{U})}{M_T} \le f(\mathbf{U}) \le f_1(\mathbf{U}) \ . \qquad (9)$$

Definition 1 Let p_1 denote the fraction of 'energy' in $\sum_t \mathbf{X}_t^2$ captured by the rank-r PCA solution $\mathbf{U}_0$. In particular,

$$p_1 = \frac{{f_1}^{\max}}{M_T} = \frac{\mathrm{Tr}\left(\mathbf{U}_0^T\left(\sum_t \mathbf{X}_t^2\right)\mathbf{U}_0\right)}{\mathrm{Tr}\left(\sum_t \mathbf{X}_t^2\right)} \ , \qquad (10)$$

so that $0 \le p_1 \le 1$.

Using this definition and Theorem 1, we have the following result bounding the value of the global maximum of CCA.

Corollary 1 *Let ${f_1}^{\max}$ and $f^{\max}$ be the global maximum of CCA1 and CCA, respectively, over $\mathbf{U}^T\mathbf{U} = \mathbb{I}_r$, and let p_1 be defined in Definition 1. Then, we have*

$$p_1 {f_1}^{\max} \le f^{\max} \le {f_1}^{\max} \ . \qquad (11)$$

Recall that the solution to CCA1 is $\mathbf{U}_0$, the top-r eigenvectors of $\sum_t \mathbf{X}_t^2$. Thus, it is easy to compute $f_1^{\max} = f_1(\mathbf{U}_0)$ and $p_1 = f_1^{\max}/M_T$. From Theorem 1, it follows that $p_1 f_1^{\max} \le f(\mathbf{U}_0) \le f_1^{\max}$. According to Corollary 1, the relative error of $f(\mathbf{U}_0)$ w.r.t. the global maximum is

$$\frac{f^{\max} - f(\mathbf{U}_0)}{f^{\max}} \le 1 - p_1 \ . \qquad (12)$$

Now if $\mathbf{U}_0$ is chosen as the initialization, the iterative updates for $f(\mathbf{U})$ converge to $\mathbf{U}_0^*$ (see Section 4) and $f(\mathbf{U}_0^*)$ satisfies

$$p_1 f_1^{\max} \le f(\mathbf{U}_0) \le f(\mathbf{U}_0^*) \le f^{\max} \le f_1^{\max} \ . \qquad (13)$$

As a result, we have the following theoretical bound for the relative error of $f(\mathbf{U}_0^*)$ w.r.t. the global maximum.

Corollary 2 *Let $\mathbf{U}_0$ be the r leading principal eigenvectors of $M(\mathbb{I}_n) = \sum_t \mathbf{X}_t^2$, and $f(\mathbf{U}_0^*)$ be the solution to CCA with the initialization $\mathbf{U}_0$. Then, the relative error of $f(\mathbf{U}_0^*)$ with respect to $f^{\max}$ satisfies*

$$\frac{f^{\max} - f(\mathbf{U}_0^*)}{f^{\max}} \le 1 - p_1 \ . \qquad (14)$$

It is interesting to note that p_1 governs the closeness of the local maximum w.r.t. the global maximum. In PCA, the fraction p_1 depends on the choice of dimension r. Empirical studies show that the upper bound $1 - p_1$ becomes fairly small for the first r leading components (see Figure 3 in Section 5.2). On the other hand, if p_1 is small ($1 - p_1$ is large), r should be increased so that a reasonable fraction (say, $p_1 = 0.9$) of energy can be preserved. The choice of r according to p_1 in CCA will be discussed in Section 3.3.

Once $f(\mathbf{U}_0^*)$ is found, the empirical bound of the relative error of $f(\mathbf{U}_0^*)$ w.r.t. the global maximum becomes

$$\frac{f^{\max} - f(\mathbf{U}_0^*)}{f^{\max}} \le 1 - \frac{f(\mathbf{U}_0^*)}{{f_1}^{\max}} \ . \qquad (15)$$

3.3 Approximate Relative Error and Rank

In certain applications, one may have to pick a suitable rank r to preserve a certain fraction of the observed covariance structure. The goal is to find the lowest rank r sufficient to explain a given fraction of the observed covariance, or equivalently to keep the approximation error below a given threshold. In PCA, since its solution based on the EVD has a nested structure, one can simply keep appending principal components until the desired error is reached. However, such a nested approximation structure is not present in CCA and more generally in the case of tensor decompositions. In particular, if the rank $(r-1)$ solution is insufficient, the computation must be carried out from scratch to obtain the rank r solution. In this section, we show that such elaborate calculations can be avoided by using the bounds relative to the CCA1 problem.

We start by defining the *Approximate Relative Error* (ARE) as a measure of quality of the the approximation obtained by CCA. For any $\mathbf{U}$, we have

$$ARE(\mathbf{U}) = \frac{\sum_{t=1}^{T} \|\mathbf{X}_t - \mathbf{U}\mathbf{Y}_t\mathbf{U}^T\|_F^2}{\sum_{t=1}^{T} \|\mathbf{X}_t\|_F^2} \ . \qquad (16)$$

We define the cumulative percentage of energy captured by the solution to CCA as follows:

Definition 2 Let $M_T = \text{Tr}(M(\mathbb{I}_n))$, and let $f(\mathbf{U}_0^*)$ be the maximum of CCA obtained by an iterative algorithm with the initialization $\mathbf{U}_0$ (see Section 4). The cumulative percent of energy p captured by $\mathbf{U}_0^*$ is defined as

$$p = \frac{f(\mathbf{U}_0^*)}{M_T} , \tag{17}$$

so that $0 \leq p \leq 1$.

For our problem, p defines how much energy over all the covariances is preserved by their corresponding latent covariances. Dividing inequality (13) by M_T and plugging in $p_1 = {f_1}^{\max}/M_T$ yield lower and upper bounds for p:

$$p_1^2 \leq p \leq p_1 . \tag{18}$$

In CCA1 (essentially PCA), given a p_1, the corresponding rank r is easy to obtain. Using the bounds for p, one can also develop a simple way to obtain a suitable rank r for CCA. To do this, we first relate p to the approximate relative error $ARE(\mathbf{U}_0^*)$.

Proposition 1 *Let $\mathbf{U}_0^*$ be the solution of CCA. Then $ARE(\mathbf{U}_0^*) = 1 - p$.*

Plugging $ARE(\mathbf{U}_0^*)$ into inequality (18), it is easy to derive the following lower and upper bounds for $ARE(\mathbf{U}_0^*)$:

$$1 - p_1 \leq ARE(\mathbf{U}_0^*) \leq 1 - p_1^2 . \tag{19}$$

Given an upper bound δ for $ARE(\mathbf{U}_0^*)$, we now show how to obtain a suitable rank r for $\mathbf{U}_0^*$ in CCA. Since $ARE(\mathbf{U}_0^*) \leq 1 - p_1^2$, it is sufficient to ensure $1 - p_1^2 \leq \delta \Rightarrow p_1 \geq \sqrt{1-\delta}$. Since p_1 corresponds to $\mathbf{U}_0$ in a PCA setting, one can easily obtain a rank-r $\mathbf{U}_0$ such that $p_1 \geq \sqrt{1-\delta}$. Initializing the iterations for CCA with $\mathbf{U}_0$ will lead to a $\mathbf{U}_0^*$ satisfying $ARE(\mathbf{U}_0^*) \leq \delta$. Note that since the construction is based on a bound, there may be a lower-rank $\mathbf{U}_0^*$ satisfying the constraint.

3.4 Conditions for Global Maximum

We now analyze a condition under which a global maximum of CCA is achieved for a given rank r. The particular case under consideration is when equality holds in (13), i.e., $f(\mathbf{U}_0^*) = f_1^{\max}$, where $f(\mathbf{U}_0^*)$ is a local maximum, implying $f(\mathbf{U}_0^*) = f^{\max}$.

Let $\mathbf{U}_0$ be the initialization consisting of the r principal eigenvectors of $M(\mathbb{I})$, we have the following result:

Theorem 2 *Let $\mathbf{U}_0$ be the r principal eigenvectors of $M(\mathbb{I}_n)$ associated with the nonzero r largest eigenvalues, then* $\text{rank}(M(\mathbf{U}_0)) \geq$ *r, where $M(\mathbf{U}_0)$ is defined in Lemma 1.*

Based on Theorem 2, we now show that $\text{rank}(M(\mathbf{U}_0)) = r$ is a necessary and sufficient condition that $f(\mathbf{U}_0^*) = f_1^{\max}$, thereby implying that $\mathbf{U}_0^*$ achieves the global optimum. Moreover, in this situation, the solution achieving the global maximum is the initialization $\mathbf{U}_0$ itself.

Theorem 3 *Let $\mathbf{U}_0$ be the solution to CCA1, i.e., the r principal eigenvectors of $M(\mathbb{I})$, and let $\mathbf{U}_0^*$ be the solution found in Algorithm 1 with the initialization* $\mathbf{U}_0$*. Then,* $\text{rank}(M(\mathbf{U}_0)) = r$ *is a necessary and sufficient condition for $f(\mathbf{U}_0^*) = f_1^{\max}$. Moreover, $\mathbf{U}_0$ is the solution achieving the global maximum for CCA.*

A special case of the result occurs when $\text{rank}(M(\mathbb{I})) = r$. When $\text{rank}(M(\mathbb{I})) = r$, $\text{rank}(M(\mathbf{U}_0)) \leq \text{rank}(M(\mathbb{I})) = r$. According to Theorem 2, $\text{rank}(M(\mathbf{U}_0)) \geq r$, implying $\text{rank}(M(\mathbf{U}_0))$ $= r$. Thus $\mathbf{U}_0$ achieves the global maximum. In this case, since all the eigenvectors are kept, the fraction of energy $p_1 = 1$. The global optimality then follows in a straightforward manner from the bounds discussed in Section 3.3.

3.5 Connections to Related Work

Given a set of rectangular matrices $\mathbf{X}_t \in \mathbb{R}^{m\times n}, 1 \leq t \leq T$, the Tucker2 model [35, 16, 24, 22], 2DSVD [9], GLRAM [37], etc., aim to find common components $\mathbf{U} \in \mathbb{R}^{m\times r}$ and $\mathbf{V} \in \mathbb{R}^{n\times s}$ such that

$$\mathbf{X}_t = \mathbf{U}\mathbf{Y}_t\mathbf{V}^T + \mathbf{E}_t , \tag{20}$$

where $\mathbf{Y}_t \in \mathbb{R}^{r\times s}$, $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices, and $\mathbf{E}_t$ is the residual. $\mathbf{U}$ and $\mathbf{V}$ can be obtained by performing the EVD iteratively on matrices $M_1(\mathbf{V}) = \sum_t \mathbf{X}_t\mathbf{V}\mathbf{V}^T\mathbf{X}_t^T \in \mathbb{R}^{m\times m}$ and $M_2(\mathbf{U}) = \sum_t \mathbf{X}_t^T\mathbf{U}\mathbf{U}^T\mathbf{X}_t \in \mathbb{R}^{n\times n}$ respectively. An initialization similar to the one proposed for CCA is usually used in these methods, e.g., initializing $\mathbf{U}$ with the EVD of $\sum_t \mathbf{X}_t\mathbf{X}_t^T$ or $\mathbf{V}$ with the EVD of $\sum_t \mathbf{X}_t^T\mathbf{X}_t$. It has been observed empirically that such an initialization usually leads to the good solutions [37, 9], particularly in rank-1 approximation experiments [27, 20]. When a locally optimal solution is found, say $(\mathbf{U}^*, \mathbf{V}^*)$, Ding et al. [9] established lower and upper bounds for a local maximum based on the eigenvalues of $M_1(\mathbf{V}^*)$ and $M_2(\mathbf{U}^*)$, but the bounds w.r.t. the global maximum were not explicitly given.

In (20), if $r = s$ and $\mathbf{Y}_t$ is diagonal, it becomes PARAFAC / CANDECOMP with orthonormal constraints [16, 17, 24, 22], referred to as PARAFAC in the sequel. If PARAFAC is applied to the covariance matrices in our case, $\mathbf{U}$ and $\mathbf{V}$ are the same. In this case, PARAFAC has the same formula as CCA except that $\mathbf{Y}_t$ is a diagonal matrix in PARAFAC but is a full matrix in CCA. In contrast to the least squares approach in PARAFAC, CPC [12, 13] simultaneously diagonalizes the positive definite matrices using a maximum likelihood approach [13]. Since the off-diagonal elements are zero in $\mathbf{Y}_t$, PARAFAC and CPC do not allow interactions among components in $\mathbf{U}$ and $\mathbf{V}$. On the other hand, if covariance matrices are simultaneously diagonalizable with an orthonormal transformation [19], it turns out that $\mathbf{Y}_t$ in CCA is also diagonal. A similar result has also been noted in the context of CPC [12, 13].

Proposition 2 *If covariance matrices $\mathbf{X}_t$ are simultaneously diagonalizable with an orthonormal transformation, the low-dimensional covariance matrices $\mathbf{Y}_t$ in CCA are diagonal.*

4. ALGORITHMS

In this section, we present algorithms for solving CCA for a given dimensionality or a given approximation error bound. For a given dimensionality, we present two algorithms that iteratively improve a given initial solution. For a given approximation error bound, we show how to determine a sufficient dimensionality, reducing the problem to the first case.

4.1 CCA for a Given Dimensionality

Iterative EVD based CCA: For a given dimensionality, EVD can be used to solve for $\mathbf{U}$ in CCA1 in (7). However, CCA in (3) has four $\mathbf{U}$'s, which cannot be found using the same approach, since this problem does not correspond to an EVD problem. Instead, we perform the EVD iteratively by fixing the two inner $\mathbf{U}$'s to the current iterate $\mathbf{U}_k$. Recall that CCA involves maximizing $f(\mathbf{U}) = \text{Tr}(\mathbf{U}^T M(\mathbf{U})\mathbf{U})$ where $M(\mathbf{U}) = \sum_{t=1}^T \mathbf{X}_t\mathbf{U}\mathbf{U}^T\mathbf{X}_t$ is of size $n \times n$. If $\mathbf{U}_k$ is the current iterate, then we compute $M(\mathbf{U}_k)$ and solve the following surrogate problem to obtain $\mathbf{U}_{k+1}$:

$$\max_{\mathbf{U}^T\mathbf{U}=\mathbb{I}_r} \text{Tr}(\mathbf{U}^T M(\mathbf{U}_k)\mathbf{U}) . \tag{21}$$

Algorithm 1 Iterative EVD (IEVD) Algorithm for CCA

```
1: Input: $\mathbf{X}_t, 1 \le t \le T$, initialization $\mathbf{U}_0 \in \mathbb{R}^{n\times r}$
2: Output: $\mathbf{U}, \mathbf{Y}_t, 1 \le t \le T$
3: repeat
4:     Perform the EVD on $M(\mathbf{U}_k) = \sum_t \mathbf{X}_t\mathbf{U}_k\mathbf{U}_k^T\mathbf{X}_t$
5:     Choose the leading $r$ eigenvectors $\mathbf{U}_{k+1}$
6:     Compute $\mathbf{Y}_t = \mathbf{U}_{k+1}^T\mathbf{X}_t\mathbf{U}_{k+1}$
7: until $\left|\frac{f(\mathbf{U}_{k+1})-f(\mathbf{U}_k)}{f(\mathbf{U}_k)}\right| \le \varepsilon$
```

Clearly, $\mathbf{U}_{k+1}$ can be obtained by applying a rank-r EVD on $M(\mathbf{U}_k)$. The idea behind such an update has been explored in the literature on tensor decompositions [24, 25, 9, 37]. As the following result shows, such an update will improve the objective function, i.e., $f(\mathbf{U}_{k+1}) \ge f(\mathbf{U}_k)$.

Theorem 4 *Let $\mathbf{U}_{k+1}$ be the r principal eigenvectors of $M(\mathbf{U}_k)$, then $f(\mathbf{U}_{k+1}) \ge \mathrm{Tr}(\mathbf{U}_{k+1}^T M(\mathbf{U}_k)\mathbf{U}_{k+1}) \ge f(\mathbf{U}_k)$. Equality holds when $\mathbf{U}_{k+1}$ and $\mathbf{U}_k$ span the same subspace.*

Algorithm 1 presents the corresponding algorithm for a given dimension r as input. The objective function increases at every step until a certain stopping criterion is satisfied. If $\mathbf{U}_0^*$ is the final solution, from the analysis of Section 3.3, we know that $f(\mathbf{U}_0^*) \ge p_1 f^{\max}$, and the approximate relative error satisfies $1 - p_1 \le ARE(\mathbf{U}_0^*) \le 1 - p_1^2$.

Auxiliary Function based CCA: In the iterative EVD method, the update has to repeatedly calculate the EVD of an $n \times n$ matrix. If n is large, the update becomes a bottleneck. In this section, we present an efficient update that only calculates the SVD of an $r \times n$ matrix. To introduce the new update, we first define an auxiliary function $g(\mathbf{U}, \mathbf{V})$ as follows:

$$g(\mathbf{U}, \mathbf{V}) = \mathrm{Tr}\left(\sum_t^T (\mathbf{U}^T\mathbf{X}_t\mathbf{U})(\mathbf{V}^T\mathbf{X}_t\mathbf{V})\right) . \qquad (22)$$

where $\mathbf{U}^T\mathbf{U} = \mathbb{I}_r$ and $\mathbf{V}^T\mathbf{V} = \mathbb{I}_r$. Clearly, $g(\mathbf{U}, \mathbf{U}) = f(\mathbf{U})$.

Given $\mathbf{U}_k$, if we can find a $\mathbf{U}_{k+1}$ satisfying $g(\mathbf{U}_k, \mathbf{U}_{k+1}) \ge g(\mathbf{U}_k, \mathbf{U}_k)$, the auxiliary function increases. Theorem 5 shows that $\mathbf{U}_{k+1}$ can be obtained by performing the SVD on an $r \times n$ matrix $\sum_t^T \mathbf{Y}_t^k\mathbf{V}^T\mathbf{X}_t$, where $\mathbf{Y}_t^k = \mathbf{U}_k^T\mathbf{X}_t\mathbf{U}_k$. Such a $\mathbf{U}_{k+1}$ increases $f(\mathbf{U})$.

Theorem 5 *Let $\mathbf{U}_{k+1} = \mathbf{Q}\mathbf{P}^T$, where $\mathbf{P}$ and $\mathbf{Q}$ are respectively the left and right r singular vectors of $\sum_t^T \mathbf{Y}_t^k\mathbf{U}_k^T\mathbf{X}_t$, where $\mathbf{Y}_t^k = \mathbf{U}_k^T\mathbf{X}_t\mathbf{U}_k$, then*

$$f(\mathbf{U}_k) \le g(\mathbf{U}_k, \mathbf{U}_{k+1}) \le f(\mathbf{U}_{k+1}) .$$

Equality holds when $\mathbf{U}_k$ and $\mathbf{U}_{k+1}$ span the same subspace.

Based on Theorem 5, we propose Algorithm 2 using the auxiliary function, yielding a solution satisfying the bounds of Section 3.3.

Time Complexity: In both algorithms, the most expensive steps are the iterative updates. In Algorithm 1, step 4 takes $O(Tn^2r)$ time to compute $M(\mathbf{U}_k)$ and $O(n^3)$ time for the EVD, and step 6 takes $O(Tn^2r)$. Overall, the cost of the updating steps in Algorithm 1 is $O(Tn^2r+n^3)$. In Algorithm 2, step 5 takes $O(Tn^2r)$ to compute the matrix $\sum_t \mathbf{Y}_t^k\mathbf{U}_k^T\mathbf{X}_t$ and $O(r^2n)$ for the SVD, and steps 6 and 7 require $O(nr^2)$ and $O(Tn^2r)$ respectively. Assuming $r \ll n$, the overall cost of the updating steps in Algorithm 2 is $O(Tn^2r)$.

Algorithm 2 Auxiliary Function (AF) Algorithm for CCA

```
1: Input: $\mathbf{X}_t, 1 \le t \le T$, initialization $\mathbf{U}_0 \in \mathbb{R}^{n\times r}$
2: Output: $\mathbf{U}, \mathbf{Y}_t, 1 \le t \le T$
3: Compute $\mathbf{Y}_t^0 = \mathbf{U}_0^T\mathbf{X}_t\mathbf{U}_0$
4: repeat
5:     Perform the SVD on matrix $\sum_t \mathbf{Y}_t^k\mathbf{U}_k^T\mathbf{X}_t = \mathbf{P}\mathbf{D}\mathbf{Q}^T$
6:     Compute $\mathbf{U}_{k+1} = \mathbf{Q}\mathbf{P}^T$
7:     Compute $\mathbf{Y}_t^{k+1} = \mathbf{U}_{k+1}^T\mathbf{X}_t\mathbf{U}_{k+1}$
8: until $\left|\frac{g(\mathbf{U}_{k+1},\mathbf{U}_{k+1})-g(\mathbf{U}_k,\mathbf{U}_k)}{g(\mathbf{U}_k,\mathbf{U}_k)}\right| \le \varepsilon$
```

4.2 CCA for a Given Approximation Error

We consider a setting where instead of the dimension r, an upper bound δ on the approximate relative error (ARE) is given. In such a setting, one can choose a sufficient dimension r and a corresponding initialization $\mathbf{U}_0$ based on our analysis in Section 3.3, and use any of the algorithms in Section 4.1 to obtain a $\mathbf{U}$ guaranteeing the error bound. In particular, it is sufficient to choose the dimension r of the initialization $\mathbf{U}_0$ such that the fraction of energy captured in CCA1 given by $p_1 = \frac{\mathrm{Tr}(\mathbf{U}_0^T M(\mathbb{I})\mathbf{U}_0)}{\mathrm{Tr}(M(\mathbb{I}))}$ satisfies $p_1 \ge \sqrt{1-\delta}$, as discussed in Section 3.3. Since $M(\mathbb{I})$ is fixed, and CCA1 is an EVD problem, choosing a suitable dimension r such that $p_1 \ge \sqrt{1-\delta}$ is straightforward due to the nested structure in the EVD. If such a $\mathbf{U}_0$ is used to initialize the algorithms in Section 4.1, the final solution $\mathbf{U}_0^*$ will satisfy $f(\mathbf{U}_0^*) \ge \sqrt{1-\delta} f^{max}$ and $ARE(\mathbf{U}_0^*) \le \delta$, which is the prescribed bound on the approximation error.

5. EXPERIMENTAL RESULTS

In this section, the performance of CCA is evaluated on both artificial datasets and two real-world stock market datasets, one spanning 21 years from 1990-2010, and the other 14 years from 1971-1984. Evaluation is done in terms of the *Approximate Relative Error* (ARE) (16) for all datasets, and also the ability to track volatility in low-dimensions for the stock market datasets. The performance of CCA is compared with PARAFAC with orthonormal constraints, PCA, and Random Projection (RP) [8, 1]. While CCA and PARAFAC are computed on the entire set of covariance matrices, PCA is computed based on the single aggregated covariance. For RP, $\mathbf{U}$ is generated as follows: (i) each entry of $\mathbf{U}$ is generated via an i.i.d. normal distribution; and (ii) $\mathbf{U}$ is normalized via the classical Gram-Schmidt orthogonalization [15] and normalization.

5.1 Artificial Data

Artificial data were generated following the model in (1). In particular, $\mathbf{Y}_t$ and $\mathbf{U}$ were generated first, then $\mathbf{X}_t$ was calculated by adding noise to $\mathbf{U}\mathbf{Y}_t\mathbf{U}^T$. $\mathbf{Y}_t$ was generated as the covariance matrix of a set of randomly generated samples. The samples were generated from the following four Gaussian distributions with means

$$m1 = [0, 0], m2 = [5, 0], m3 = [0, 5], m4 = [5, 5] ,$$

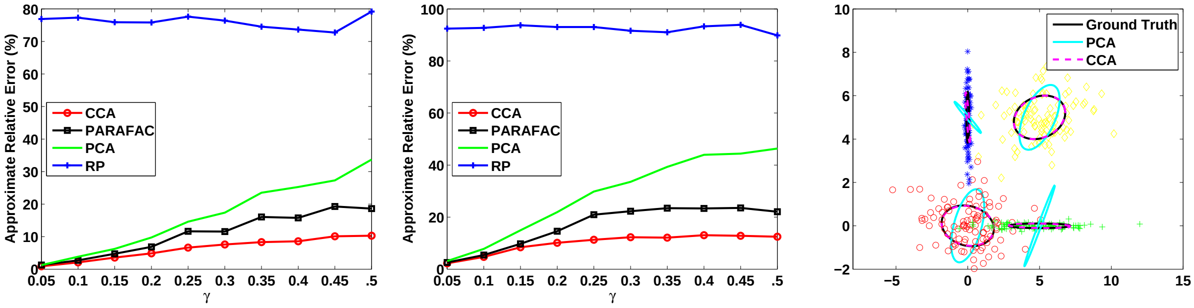
and covariances

$$[\mathbf{\Sigma}_1|\mathbf{\Sigma}_2|\mathbf{\Sigma}_3|\mathbf{\Sigma}_4] = \left[\begin{array}{cc|cc|cc|cc} 4 & 0 & 4 & 0 & 0.01 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0.01 & 0 & 2 & 0 & 2 \end{array}\right] .$$

Instead of using a fixed $\mathbf{U}$, it was mildly perturbed as follows:

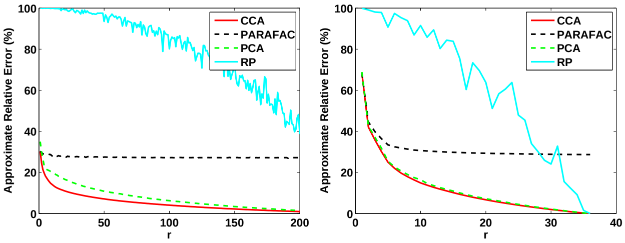
$$\mathbf{U}_{t+1} \leftarrow QR(\mathbf{U}_t + \gamma E_t) , \qquad (23)$$

where γ is a small constant, $E_t \in \mathbb{R}^{n\times r}$ where $E_{ij} \sim N(0, 1)$, and $\mathbf{U}_{t+1}$ is obtained from the QR factorization of $(\mathbf{U}_t + \gamma E_t)$. In (23), $\mathbf{U}_1$ was randomly generated, $r = 2$, and we considered two values of the high-dimensionality $n = 5, 10$. The experiment was



(a) $n = 5$ (b) $n = 10$ (c) Latent Covariances in 2D

Figure 1: (a)-(b) Approximation Relative Error (ARE) on artificial data in different dimensions r and increasing noise level γ. CCA outperforms PARAFAC, PCA and RP, especially with high noise levels. (c) 2D latent covariances. CCA tracks the true covariance better than PCA.



(a) S&P500 (b) NYSE

Figure 2: Approximation Relative Error (ARE) on S&P 500 and NYSE in different dimensions r. CCA outperforms PARAFAC, PCA and RP.

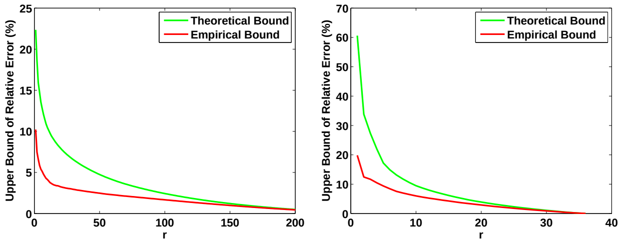
repeated 50 times, and the final results reported were the average over the 50 runs.

Results: Figure 1 (a)-(b) shows the comparative performance of CCA, PARAFAC, PCA, and RP in terms of the ARE (lower is better) across different noise levels γ for fixed low-dimensionality $r = 2$. As the figures show, CCA outperforms PARAFAC and PCA, and significantly outperforms RP. The improvement of CCA over other methods is more pronounced for high noise levels (high γ). For low noise levels, CCA and PARAFAC are competitive since all the covariance matrices are nearly diagonal. Due to the structure of the covariance matrices (nearly diagonal but different), PARAFAC outperforms PCA which maximizes the total covariance instead.

Figure 1(c) shows the shape of 2-dimensional covariances when $n = 10, \gamma = 0.1$. For each Gaussian distribution, covariance of the samples is its ground truth, which is plotted in black. The latent covariances for CCA and PCA, shown respectively in magenta and cyan, are calculated based on the leading 2 components. Since the black ellipses are entirely overlapped by the red ones, the ground truth is not visible. While CCA is able to recover the ground truth, PCA seems to find a subspace that maximizes the total covariance but is not suitable for separate covariances.

5.2 Stocks Data

We considered two real world stock market datasets. The first dataset, S&P500, is based on daily closing prices of the 263 stocks in the current S&P500 index from 1990 to 2010. The second dataset, NYSE, is a widely used dataset of daily closing prices of 36 stocks at daily resolution spanning from 1971 to 1984 [18, 2, 7].



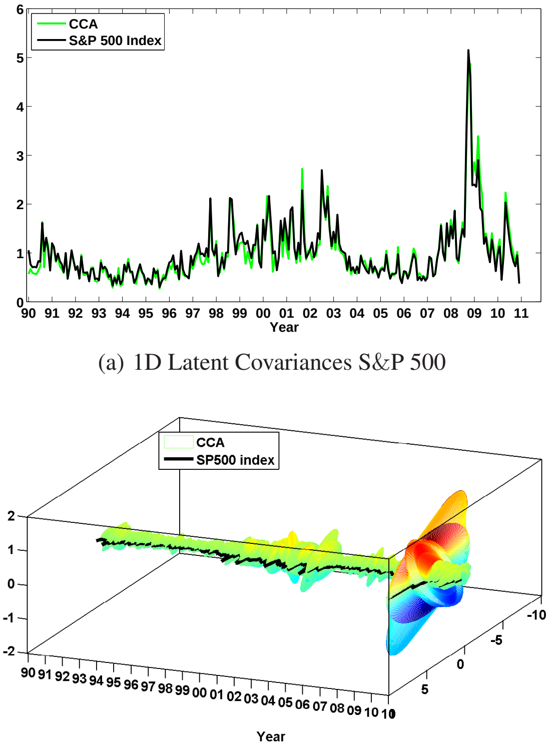
(a) S&P500 (b) NYSE

Figure 3: The upper bound of the relative error of the CCA results w.r.t. the global maximum on S&P 500 and NYSE in different dimensions r. The theoretical bound in (14) and empirical bound in (15) are in green and red, respectively. The upper bounds are very small for the first r principal components.

Methodology: For the experiments, the covariance of the daily log-return was considered for both datasets, where return$_t$ = $\log \frac{x_t}{x_{t-1}} \times 100\%$, x_t is the daily closing stock price. For each dataset, we constructed the monthly average of the daily covariances, and each average monthly covariance was considered as an observed covariance matrix $\mathbf{X}_t$. For S&P500, there are $21 \times 12 = 252$ observed covariance matrices $\mathbf{X}_t \in \mathbb{R}^{263\times263}$. For NYSE, there are $14 \times 12 = 168$ covariance matrices $\mathbf{X}_t \in \mathbb{R}^{36\times36}$.

ARE: The performance of the four methods is evaluated in terms of the ARE on S&P500 and NYSE, as shown in Figure 2. On both datasets, CCA outperforms PARAFAC and PCA, and significantly outperforms RP. Interestingly, the performance of PARAFAC does not improve with increasing r (dimensionality) possibly because the covariances cannot be simultaneously diagonalized. PCA performs much better than PARAFAC, which is in direct contrast with the observed results for the artificial dataset. Note that CCA performs the best on both types of data, which illustrates the flexibility of the model. PCA is competitive with CCA on NYSE but worse on S&P500, especially for low dimensions. There are two possible explanations: NYSE is a low-dimensional dataset with $n = 36$, whereas S&P500 is a relatively high-dimensional dataset with $n = 263$; and the stock market has been more volatile in the 1990-2010 range (S&P500) as compared to the 1971-1984 range (NYSE).

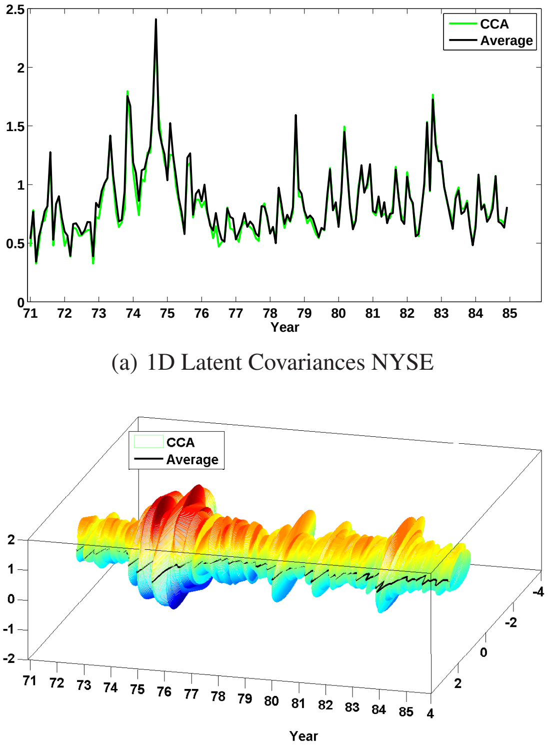
Quality of CCA Solution: Figure 3 shows the upper bounds of the relative error of CCA results w.r.t. the global maximum value. The



(a) 1D Latent Covariances S&P 500

(b) 2D Latent Covariances S&P 500

Figure 4: Latent Covariances over time for S&P500 from 1990 to 2010. The two financial meltdowns in 2001 and 2008 are prominently captured in the latent low-dimensional space.



(a) 1D Latent Covariances NYSE

(b) 2D Latent Covariances NYSE

Figure 5: Latent Covariances over time for NYSE from 1970 to 1984. The stock market crash of 1974 is captured in the latent low-dimensional space.

green curve is the theoretical upper bound $1 - p_1$ in (14), which depends on the choice of dimensionality r and the dataset. When $r = 1$, the theoretical bound is fairly good on S&P500 but bad on NYSE. However, as r increases, the bounds decrease rapidly. When $r = 10$, the theoretical upper bounds are already approximately 10% on both datasets. After a local maximum $f(\mathbf{U}_0^*)$ is found, the empirical upper bound in (15) is plotted in red. On both datasets, the empirical upper bounds provide significant improvements over the theoretical upper bounds, especially for low values of r. For example, the empirical upper bound on NYSE decrease to 20% when $r = 1$. When $r = 10$, the empirical bounds further decrease to 5% on both datasets.

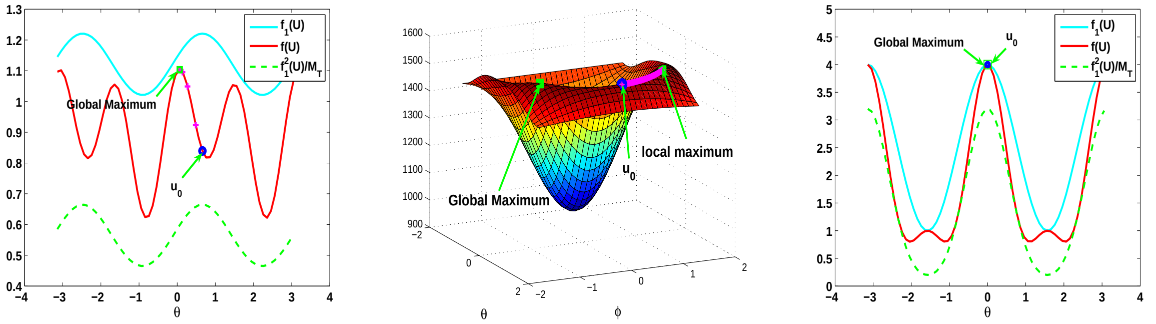
Volatility: In Figures 4 and 5 we plot the latent covariance matrices (level sets) obtained from CCA in dimensions $r = 1, 2$ for S&P500 and NYSE, and compare them to the volatilities [5, 11, 10] of their proxies. The proxy of the S&P 500 dataset is the S&P500 index, while the proxy of NYSE is the average of 36 stocks. The reason we expect $\mathbf{Y}_t$ to track volatility well is as follows: For n stocks, the trace of the covariance $\mathbf{X}_t$ is equal to $n\sigma^2$, where σ is the volatility (standard deviation) of the proxy. If $\mathbf{Y}_t$ approximates $\mathbf{X}_t$ well, the trace of $\sqrt{\mathbf{Y}_t/n}$ should approximate σ. In both datasets, for 1D ($r = 1$), $\sqrt{\mathbf{Y}_t/n}$ tracks the volatility almost exactly. For 2D ($r = 2$), $\sqrt{\mathbf{Y}_t/n}$ are ellipses that change shape/size over time, and the volatility (black curve) is always on the circumference of the ellipses. It is interesting to note that the latent covariances for S&P 500 (Figure 4) seem to capture the two major financial meltdowns, viz the dot-com bubble around 2001 and the major financial crisis around 2008, even in such a low dimensionality. The crisis in 2008 looks significantly worse, and the ellipses in the 2D plot have different shapes possibly indicating different market segments being more adversely affected. Similarly, the latent covariances for NYSE (Figure 5) capture the stock market crash around 1973-1974 resulting from the collapse of the Bretton Woods system along with the 'Nixon Shock' and the devaluation of the US dollar. Such interpretable results show the potential of CCA in high-dimensional real world problems.

Choose r given ARE: We also evaluated our method in choosing the dimensionality r given an ARE upper bound. The results on the S&P 500 dataset are shown in Table 2. The first row is the given ARE upper bound δ, the second row shows the sufficient r computed as in Section 4.2 and the corresponding ARE, and the third row shows the smallest r that would have satisfied the bound and the corresponding ARE. The chosen r satisfies the bound, but can be conservative at times especially when the ARE decreases rapidly with increasing r.

$\delta(\%)$	30	20	10	5
Chosen r (ARE)	3(21.50)	10(14.18)	45(7.58)	97(4.20)
Smallest r (ARE)	2(24.67)	4(19.70)	26(9.88)	81(5.00)

Table 2: Choosing r given an ARE upper bound on S&P 500.

Running Time: Figure 6 compares the running times (in seconds) of Algorithm 1 and Algorithm 2 on the S&P 500 dataset. The experiments were run in Matlab 7.1 on an Intel P8600 2.4GHz PC with 2G memory. When r is small, i.e., low-dimensional projections, Algorithm 2 is much faster than Algorithm 1. As r increases, Algorithm 2 possibly spends more time on the SVD step, and probably requires more steps to converge, so the superiority in running time decreases.



(a) A global maximum is found with the proposed initialization (b) A local maximum is found with the proposed initialization (c) A global maximum is the initialization

Figure 7: Optimizing $f(\mathrm{U})$ in CCA based on CCA1 initialization and iterative updates. Objective $f(\mathrm{U})$ for CCA is shown in red; the lower and upper bounds based on $f_1(\mathrm{U})$ for CCA1 are shown in green and cyan respectively. Three scenarios are shown: (a) iterations converge to a global maximum, (b) iterations converge to a local maximum, and (c) initialization is a global maximum.

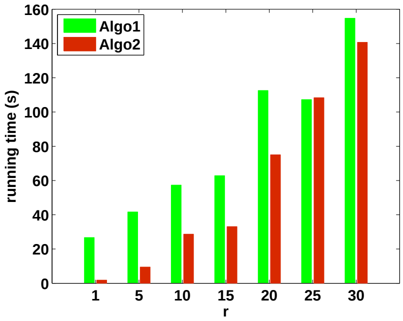


Figure 6: Running times of Algorithm 1 and Algorithm 2 on S&P 500 in different dimensions r. The auxiliary function based method (Algorithm 2) is distinctly faster for low-dimensional projections.

5.3 Additional Numerical Simulations

We study CCA (Algorithm 1) on low-dimensional problems to better understand the proposed ideas, including cases where the approach can and cannot find the global maximum of $f(\mathbf{U})$. It is important to recall that while $f(\mathbf{U})$ is a convex function for unconstrained $\mathbf{U}$, the model requires maximizing $f(\mathbf{U})$ on the domain of $\mathbf{U}$ determined by $\mathbf{U}^T\mathbf{U} = \mathbb{I}_r$, and the problem may thus have multiple local maxima.

We illustrate different scenarios for using Algorithm 1 to solve CCA in Figure 7. In Figure 7(a), we consider 3 time steps for a 2-dimensional covariance matrix, with

$$\mathbf{X} = [\mathbf{X}_1|\mathbf{X}_2|\mathbf{X}_3] = \left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0.22 & 0.22 \\ 0 & 0.25 & 0 & 1 & 0.22 & 0.22 \end{array}\right].$$

The vector $\mathbf{u}$ is parameterized as $\mathbf{u} = [\sin(\theta), \cos(\theta)]^T$, and the x-axis denotes θ. Note that $f(\mathbf{u})$ is convex in $\mathbf{u}$ but not θ, which explains the nonconvex plot of the objective (in red). Further, the domain of θ is in $[-\pi, \pi]$, and the function is periodic beyond that domain. Algorithm 1 is used to find the best rank-1 approximation $\mathbf{u}$. In particular, the initialization $\mathbf{u}_0$ is the optimal solution of $f_1(\mathbf{u})$, denoted by a small blue circle $\circ$. The searching trajectory is denoted by magenta +, and the optimal solution of $f(\mathbf{u})$ by a green $\square$. The upper and lower bounds are plotted in cyan and green respectively. For this scenario, with the proposed initialization, a global maximum can be found, as illustrated in Figure 7(a). However, the initialization does not always lead to a global maximum as shown in Figure 7(b). In Figure 7(b), we consider

$$\mathbf{X}_1 = \begin{bmatrix} 29.7995 & 2.5707 & 1.7377 \\ 2.5707 & 30.1445 & -0.0292 \\ 1.7377 & -0.0292 & 24.1799 \end{bmatrix},$$

$$\mathbf{X}_2 = \begin{bmatrix} 21.8515 & -2.2068 & 2.0377 \\ -2.2068 & 22.8371 & 0.0490 \\ 2.0377 & 0.0490 & 21.1336 \end{bmatrix},$$

$$\mathbf{X}_3 = \begin{bmatrix} 8.5273 & -2.5322 & 1.1011 \\ -2.5322 & 9.6724 & -0.9796 \\ 1.1011 & -0.9796 & 6.4754 \end{bmatrix},$$

and the vector $\mathbf{u}$ is parameterized as $\mathbf{u} = [\sin(\theta), \cos(\theta)\sin(\phi), \cos(\theta)\cos(\phi)]^T$. In Figure 7(b), θ and ϕ are the x-axis and y-axis respectively, and $f(\mathbf{u})$ is shown in the z-axis. For this scenario, the final solution is a good local maximum but is not a global maximum, which is also marked in the figure. Finally, Figure 7(c) shows a case where the initialization itself achieves a global maximum of CCA. In Figure 7(c), we consider

$$\mathbf{X} = [\mathbf{X}_1|\mathbf{X}_2] = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right],$$

and $\mathbf{u}$ is parameterized as in Figure 7(a). For this scenario, if $\mathbf{u}_0$ denotes the initialization obtained from CCA1, we see that $f_1^{\max} = f_1(\mathbf{u}_0) = f(\mathbf{u}_0)$, implying $f(\mathbf{u}_0) = f^{\max}$.

6. CONCLUSIONS

In this paper, we introduced a framework called CCA for simultaneously modeling multiple covariance matrices in low dimensions. While the framework has similarities with existing approaches to tensor decompositions, we presented a novel and unique analysis of CCA in terms of a more tractable PCA framework called CCA1, which provides lower and upper bounds for the global maximum for CCA. The bounds also lead to an effective initialization scheme so that the results of CCA have clear approximation guarantees w.r.t. the global maximum. We also discussed non-trivial conditions under which the global maximum will be achieved. We proposed two algorithms: a standard tensor decomposition algorithm and an efficient auxiliary function based algorithm. They can work with either a fixed dimensionality or an approximate relative error. We illustrated the effectiveness of our approaches on synthetic data and on two real world stock market datasets.

While CCA can be considered as a special case of classical tensor decomposition methods, the analysis presented in this paper discusses two important issues encountered in the general case. Such an analysis can potentially be extended to more general settings considered in the tensor decomposition literature, and will be considered in future work. In the present analysis, all covariance matrices were assumed to be available. In real life domains such as finance and climate sciences, the observed covariance matrices become available over time. We plan to investigate extensions of the CCA framework to the online setting where the observed matrices become available over time.

Acknowledgment

This research was supported by NSF grants IIS-0916750, IIS-0812183, IIS-0534286, NSF CAREER grant IIS-0953274 and NASA grant NNX08AC36A.

7. REFERENCES

[1] D. Achlioptas. Database-friendly random projections. In *ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 274–281, 2001.

[2] A. Agarwal, E. Hazan, S. Kale, and R. E. Schapire. Algorithms for portfolio management based on the Newton method. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 2006.

[3] T. Anderson. *An Introduction to Multivariate Statistics, 3rd ed.* John Wiley, 2003.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

[5] T. Bollerslev, J. Russell, and M. Watson. *Volatility and Time Series Econometrics: Essays in Honor of Robert Engle*. 2010.

[6] D. Cai, X. He, and J. Han. Subspace learning based on tensor analysis. In *Technical Report UIUCDCS-R-2005-2572*, 2005.

[7] T. M. Cover. Universal portfolios. *Mathematical Finance*, 1(1):1–29, 1991.

[8] S. Dasgupta. Experiments with random projection. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 143–151, 2000.

[9] C. Ding and J. Ye. Two-dimensional singular value decomposition (2DSVD) for 2D maps and images. In *Proceedings of the 5th SIAM International Conference on Data Mining (SDM)*, pages 32–43, 2005.

[10] R. Engle. Autoregressive conditional heteroskedasticity with estimates of the variance of UK inflation. *Econometrica*, 50:987–1008, 1982.

[11] R. Engle. Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroscedasticity models. *Journal of Business and Economic Statistics*, 20:339–350, 2002.

[12] B. N. Flury. Common principal components in k groups. *Journal of American Statistical Association*, 79(388):892–898, 1984.

[13] B. N. Flury. *Common Principal Components and Related Multivariate Models*. John Wiley, 1988.

[14] K. Fukunaga. *Introduction to Statistical Pattern Recognition, 2nd edition*. Academic Press, 1990.

[15] G. H. Golub and C. V. Loan. *Matrix Computations,3rd ed.* Johns Hopkins University Press, 1996.

[16] R. A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an explanatory multimodal factor analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970.

[17] R. A. Harshman. PARAFAC. Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems*, 38(2):149–171, 1997.

[18] D. Helmbold, R. Schapire, Y. Singer, and M. Warmuth. Online portfolio setection using multiplicative weights. *Mathematical Finance*, 8(4):325–347, 1998.

[19] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[20] E. Kofidis and P. A. Regalia. On the best rank-1 approximation of higher-order supersymmetric tensors. *SIAM Journal on Matrix Analysis and Applications*, 23(3):863–884, 2000.

[21] T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.

[22] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[23] T. G. Kolda, B. W. Bader, and J. P. Kenny. Higher- order web link analysis using multilinear algebra. In *Proceedings of the fifth IEEE International Conference on Data Mining (ICDM)*, pages 242–249, 2005.

[24] P. M. Kroonenberg. *Applied Multiway Data Analysis*. Wiley, 2008.

[25] P. M. Kroonenberg and J. de Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45(1):69–97, 1980.

[26] L. D. Lathauwer, B. D. Moor, J. Vandewalle, and J. V. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[27] L. D. Lathauwer, B. D. Moor, J. Vandewalle, and J. V. On the best rank-1 and rank-$(r_1, r_2, ..., r_n)$ approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342, 2000.

[28] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 556–562, 2001.

[29] J. D. Leeuw and G. Michaildis. Majorization methods in statistics. *Biostatistics*, 9(3):432–441, 2008.

[30] J. A. Patz, D. Campdell-Lendrum, T. Holloway, and J. A. Foley. Impact of regional climate change on human health. *Nature*, 438:310–317, 2005.

[31] M. Pourahmadi, M. J. Daniels, and T. Park. Simultaneous modelling of the Cholesky decomposition of several covariance matrices. *Journal of Multivariate Analysis*, 98:568–587, 2006.

[32] J. T. Scruggs and P. Glabadanidis. Risk premia and the dynamic covariance between stock and bond returns. *Journal of finance and quantitative analysis*, 38(2):295–316, 2003.

[33] J. Sun, D. Tao, and C. Faloutsos. Beyond streams and graphs: Dynamic tensor analysis. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 374–383, 2006.

[34] S. Tadjudin and D. A. Landgrebe. Covariance estimation with limited training samples. *IEEE Transactions on Geoscience and Remote Sensing*, 37(4), 1999.

[35] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

[36] H. Wang, A. Banerjee, and D. Boley. Common component analysis for multiple covariance matrices. *Technical Report, TR-10-017, University of Minnesota, Twin Cities*, 2010. `http://www.cs.umn.edu/tech_reports_upload/tr2010/10-017.pdf`.

[37] J. Ye. Generalized low rank approximations of matrices. *Machine Learning Journal*, 61:167–191, 2005.