Stay on path: PCA along graph paths

Megasthenis Asteris MEGAS@UTEXAS.EDU
Anastasios Kyrillidis ANASTASIOS@UTEXAS.EDU
Alexandros G. Dimakis DIMAKIS@AUSTIN.UTEXAS.EDU
Department of Electrical and Computer Engineering, The University of Texas at Austin

Han-Gyol Yi GYOL@UTEXAS.EDU
Bharath Chandrasekaran BCHANDRA@AUSTIN.UTEXAS.EDU
Department of Communication Sciences & Disorders, The University of Texas at Austin

Abstract

We introduce a variant of (sparse) PCA in which the set of feasible support sets is determined by a graph. In particular, we consider the following setting: given a directed acyclic graph G on p vertices corresponding to variables, the non-zero entries of the extracted principal component must coincide with vertices lying along a path in G.

From a statistical perspective, information on the underlying network may potentially reduce the number of observations required to recover the population principal component. We consider the canonical estimator which optimally exploits the prior knowledge by solving a non-convex quadratic maximization on the empirical covariance. We introduce a simple network and analyze the estimator under the spiked covariance model. We show that side information potentially improves the statistical complexity.

We propose two algorithms to approximate the solution of the constrained quadratic maximization, and recover a component with the desired properties. We empirically evaluate our schemes on synthetic and real datasets.

1. Introduction

Principal Component Analysis (PCA) is an invaluable tool in data analysis and machine learning. Given a set of n centered p-dimensional datapoints $\mathbf{Y} \in \mathbb{R}^{p \times n}$, the first principal component is

$$\arg\max_{\|\mathbf{x}\|_2=1} \mathbf{x}^\top \widehat{\boldsymbol{\Sigma}} \mathbf{x}, \tag{1}$$

where $\widehat{\boldsymbol{\Sigma}} = 1/n \cdot \mathbf{Y}\mathbf{Y}^\top$ is the empirical covariance matrix. The principal component spans the direction of maximum data variability. This direction usually involves all p variables of the ambient space, in other words the PC vectors are typically non-sparse. However, it is often desirable to obtain a principal component with specific structure, for example limiting the support of non-zero entries. From a statistical viewpoint, in the high dimensional regime $n = O(p)$, the recovery of the true (population) principal component is only possible if additional structure information, like sparsity, is available for the former (Amini & Wainwright, 2009; Vu & Lei, 2012).

There are several approaches for extracting a sparse principal component. Many rely on approximating the solution to

$$\max_{\mathbf{x} \in \mathbb{R}^p} \mathbf{x}^\top \widehat{\boldsymbol{\Sigma}} \mathbf{x}, \quad \text{subject to } \|\mathbf{x}\|_2 = 1,\ \|\mathbf{x}\|_0 \leq k. \tag{2}$$

The non-convex quadratic optimization is NP hard (by a reduction from maximum clique problem), but optimally exploits the side information on the sparsity.

Graph Path PCA. In this paper we enforce additional structure on the support of principal components. Consider a directed acyclic graph (DAG) $G = (V, E)$ on p vertices. Let S and T be two additional special vertices and consider all simple paths from S to T on the graph G. Ignoring the order of vertices along a path, let $\mathcal{P}(G)$ denote the collection of all S-T paths in G. We seek the principal component supported on a path of G, *i.e.*, the solution to

$$\max_{\mathbf{x} \in \mathcal{X}(G)} \mathbf{x}^\top \widehat{\boldsymbol{\Sigma}} \mathbf{x}, \tag{3}$$

where

$$\mathcal{X}(G) \triangleq \left\{ \mathbf{x} \in \mathbb{R}^p : \|\mathbf{x}\|_2 = 1,\ \text{supp}(\mathbf{x}) \in \mathcal{P}(G) \right\}. \tag{4}$$

We will argue that this formulation can be used to impose several types of structure on the support of principal components. Note that the covariance matrix $\widehat{\boldsymbol{\Sigma}}$ and the graph

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

can be arbitrary: the matrix is capturing data correlations while the graph is a mathematical tool to efficiently describe the possible supports of interest. We illustrate this through a few applications.

Financial model selection: Consider the problem of identifying which companies out of the S&P500 index capture most data variability. Running Sparse PCA with a sparsity parameter k will select k companies that maximize explained variance. However, it may be useful to enforce more structure: If we must select one company from each business sector (*e.g.*, Energy, Health Care,*etc.*) *how could we identify these representative variables?*

In Section 4, we show that this additional requirement can be encoded using our graph path framework. We compare our variable selection with Sparse PCA and show that it leads to interpretable results.

Biological and fMRI networks: Several problems involve variables that are naturally connected in a network. In these cases our Graph Path PCA can enforce interpretable sparsity structure, especially when the starting and ending points are manually selected by domain experts. In section 4, we apply our algorithm on fMRI data using a graph on regions of interest (ROIs) based on the Harvard-Oxford brain structural atlas (Desikan et al., 2006).

We emphasize that our applications to brain data is preliminary: the directional graphs we extract are simply based on distance and should not be interpreted as causality, simply as a way of encoding desired supports.

What can we say about the tractability of (3)*?* We note that despite the additional constraints on the sparsity patterns, the number of admissible support sets (*i.e.* S-T paths) can be exponential in p, the number of variables. For example, consider a graph G as follows: S connected to two nodes who are then both connected to two nodes, *etc.* for k levels and finally connected to T. Clearly there are 2^k S-T paths and therefore a direct search is not tractable.

Our Contributions:

1. From a statistical viewpoint, we show that side information on the underlying graph G can reduce the number of observations required to recover the population principal component $\mathbf{x}_\star \in \mathcal{X}(G)$ via (3). For our analysis, we introduce a simple, sparsity-inducing network model on p vertices partitioned into k layers, with edges from one layer to the next, and maximum out-degree d (Fig. 1). We show that $n = O(\log p/k + k \log d)$ observations $\mathbf{y}_i \sim N(\mathbf{0}, \mathbf{\Sigma})$, suffice to obtain an arbitrarily good estimate via (3). Our proof follows the steps of (Vu & Lei, 2012).

2. We complement this with an information-theoretic lower bound on the minimax estimation error, under the spiked covariance model with latent signal $\mathbf{x}_\star \in \mathcal{X}(G)$, which matches the upper bound.

3. We propose two algorithms for approximating the solution of (3), based on those of (Yuan & Zhang, 2013) and (Papailiopoulos et al., 2013; Asteris et al., 2014) for the sparse PCA problem. We empirically evaluate our algorithms on synthetic and real datasets.

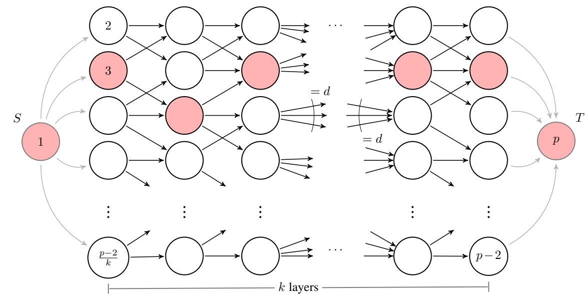

Figure 1. A (p, k, d)-layer graph $G = (V, E)$: a DAG on p vertices, partitioned into k disjoint sets (layers) $\mathcal{L}_1, \ldots, \mathcal{L}_k$. The highlighted vertices form an S-T path.

Related Work There is a large volume of work on algorithms and the statistical analysis of sparse PCA (Johnstone & Lu, 2004; Zou et al., 2006; d'Aspremont et al., 2008; 2007; Johnstone & Lu, 2004; Vu & Lei, 2012; Amini & Wainwright, 2009). On the contrary, there is limited work that considers additional structure on the sparsity patterns. Motivated by a face recognition application, (Jenatton et al., 2010) introduce *structured sparse PCA* using a regularization that encodes higher-order information about the data. The authors design sparsity inducing norms that further promote a pre-specified set of sparsity patterns.

Finally, we note that the idea of pursuing additional structure on top of sparsity is not limited to PCA: Model-based compressive sensing seeks sparse solutions under a restricted family of sparsity patterns (Baldassarre et al., 2013; Baraniuk et al., 2010; Kyrillidis & Cevher, 2012), while structure induced by an underlying network is found in (Mairal & Yu, 2011) for sparse linear regression.

2. A Data Model – Sample Complexity

The layer graph. Consider a directed acyclic graph $G = (V, E)$ on p vertices, with the following properties:

- $V = \{S, T\} \cup \widehat{V}$, where S is a source vertex, T is a terminal one, and $\widehat{V}$ is the set of remaining $p - 2$ vertices.
- $\widehat{V}$ can be partitioned into k disjoint subsets (*layers*) $\mathcal{L}_1, \ldots, \mathcal{L}_k$, *i.e.*, $\bigcup_i \mathcal{L}_i = \widehat{V}$, and $\mathcal{L}_i \cap \mathcal{L}_j = \emptyset$, $\forall i, j \in [k]$, $i \neq j$, such that:
  - $\Gamma_{\text{out}}(v) \subset \mathcal{L}_{i+1}$, $\forall v \in \mathcal{L}_i$, for $i = 1, \ldots, k-1$, where $\Gamma_{\text{out}}(v)$ denotes the out-neighborhood of v.

– $\Gamma_{\text{out}}(S) = \mathcal{L}_1$, and $\Gamma_{\text{out}}(v) = \{T\}, \forall v \in \mathcal{L}_k$.

In the sequel, for simplicity, we will further assume that $p-2$ is a multiple of k and $|\mathcal{L}_i| = {(p-2)}/{k}, \forall i \in [k]$. Further, $|\Gamma_{\text{out}}(v)| = d, \forall v \in \mathcal{L}_i, i = 1, \ldots, k-1$, and $|\Gamma_{\text{in}}(v)| = d, \forall v \in \mathcal{L}_i, i = 2, \ldots, k$, where $\Gamma_{\text{in}}(v)$ denotes the in-neighborhood of v. In words, the edges from one layer are maximally spread accross the vertices of the next. We refer to G as a (p,k,d)-*layer graph*.

Fig. 1 illustrates a (p,k,d)-layer graph G. The highlighted vertices form an S-T path π: a *set* of vertices forming a trail from S to T. Let $\mathcal{P}(G)$ denote the collection of S-T paths in a graph G for a given pair of source and terminal vertices. For the (p,k,d)-layer graph, $|\pi| = k, \forall \pi \in \mathcal{P}(G)$, and

$$|\mathcal{P}(G)| = |\mathcal{L}_1| \cdot d^{k-1} = \tfrac{p-2}{k} \cdot d^{k-1} \leq \tbinom{p-2}{k},$$

since $d \in \{1, \ldots, {(p-2)}/{k}\}$.

Spike along a path. We consider the *spiked covariance model*, as in the sparse PCA literature (Johnstone & Lu, 2004; Amini & Wainwright, 2008). Besides sparsity, we impose additional structure on the latent signal; structure induced by a (known) underlying graph G.

Consider a p-dimensional signal $\mathbf{x}_\star$ and a bijective mapping between the p variables in $\mathbf{x}_\star$ and the vertices of G. For simplicity, assume that the vertices of G are labeled so that x_i is associated with vertex $i \in V$. We restrict $\mathbf{x}_\star$ in

$$\mathcal{X}(G) \triangleq \big\{\mathbf{x} \in \mathbb{R}^p : \|\mathbf{x}\|_2 = 1,\ \mathrm{supp}(\mathbf{x}) \in \mathcal{P}(G)\big\},$$

that is, $\mathbf{x}_\star$ is a unit-norm vector whose active (nonzero) entries correspond to vertices along a path in $\mathcal{P}(G)$.

We observe n points (samples) $\{\mathbf{y}_i\}_{i=1}^n \in \mathbb{R}^p$, generated randomly and independently as follows:

$$\mathbf{y}_i = \sqrt{\beta} \cdot u_i \cdot \mathbf{x}_\star + \mathbf{z}_i, \tag{5}$$

where the scaling coefficient $u_i \sim \mathcal{N}(0,1)$ and the additive noise $\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p)$ are independent. Equivalently, $\mathbf{y}_i$s are *i.i.d.* samples, distributed according to $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, where

$$\mathbf{\Sigma} = \mathbf{I}_p + \beta \cdot \mathbf{x}_\star \mathbf{x}_\star^\top. \tag{6}$$

2.1. Lower bound

Theorem 1 (Lower Bound). *Consider a (p,k,d)-layer graph G on p vertices, with $k \geq 4$, and $\log d \geq 4H(3/4)$. (Note that $p-2 \geq k \cdot d$), and a signal $\mathbf{x}_\star \in \mathcal{X}(G)$. Let $\{\mathbf{y}_i\}_{i=1}^n$ be a sequence of n random observations, independently drawn according to probability density function*

$$\mathcal{D}_p(\mathbf{x}_\star) = \mathcal{N}\big(\mathbf{0}, \mathbf{I}_p + \beta \cdot \mathbf{x}_\star \mathbf{x}_\star^\top\big),$$

for some $\beta > 0$. Let $\mathcal{D}_p^{(n)}(\mathbf{x}_\star)$ denote the product measure over the n independent draws. Consider the problem of estimating $\mathbf{x}_\star$ from the n observations, given G. There exists $\mathbf{x}_\star \in \mathcal{X}(G)$ such that for every estimator $\widehat{\mathbf{x}}$,

$$\mathbb{E}_{\mathcal{D}_p^{(n)}(\mathbf{x}_\star)}\big[\|\widehat{\mathbf{x}}\widehat{\mathbf{x}}^\top - \mathbf{x}_\star \mathbf{x}_\star^\top\|_{\text{F}}\big] \geq \frac{1}{2\sqrt{2}} \cdot \sqrt{\min\Big\{1,\ \tfrac{C' \cdot (1+\beta)}{\beta^2} \cdot \tfrac{1}{n}\left(\log \tfrac{p-2}{k} + \tfrac{k}{4}\log d\right)\Big\}}. \tag{7}$$

Theorem 1 effectively states that for some latent signal $\mathbf{x}_\star \in \mathcal{X}(G)$, and observations generated according to the spiked covariance model, the minimax error is bounded away from zero, unless $n = \Omega\left(\log {p}/{k} + k \log d\right)$. In the sequel, we provide a sketch proof of Theorem 1, following the steps of (Vu & Lei, 2012).

The key idea is to discretize the space $\mathcal{X}(G)$ in order to utilize the Generalized Fano Inequality (Yu, 1997). The next lemma summarizes Fano's Inequality for the special case in which the n observations are distibuted according to the n-fold product measure $\mathcal{D}_p^{(n)}(\mathbf{x}_\star)$:

Lemma 2.1 (Generalized Fano (Yu, 1997))**.** *Let $\mathcal{X}_\epsilon \subset \mathcal{X}(G)$ be a finite set of points $\mathbf{x}_1, \ldots, \mathbf{x}_{|\mathcal{X}_\epsilon|} \in \mathcal{X}(G)$, each yielding a probability measure $\mathcal{D}_p^{(n)}(\mathbf{x}_i)$ on the n observations. If $d(\mathbf{x}_i, \mathbf{x}_j) \geq \alpha$, for some pseudo-metric[1] $d(\cdot,\cdot)$ and the Kullback-Leibler divergences satisfy*

$$\mathrm{KL}\big(\mathcal{D}_p^{(n)}(\mathbf{x}_i)\ \|\ \mathcal{D}_p^{(n)}(\mathbf{x}_j)\big)\ \leq\ \gamma,$$

for all $i \neq j$, then for any estimator $\widehat{\mathbf{x}}$

$$\max_{\mathbf{x}_i \in \mathcal{X}_\epsilon} \mathbb{E}_{\mathcal{D}_p^{(n)}(\mathbf{x}_i)}[d(\widehat{\mathbf{x}}, \mathbf{x}_i)]\ \geq\ \frac{\alpha}{2} \cdot \left(1 - \frac{\gamma + \log 2}{\log|\mathcal{X}_\epsilon|}\right). \tag{8}$$

Inequality (8), using the pseudo-metric

$$d\left(\widehat{\mathbf{x}}, \mathbf{x}\right)\ \triangleq\ \|\widehat{\mathbf{x}}\widehat{\mathbf{x}}^\top - \mathbf{x}\mathbf{x}^\top\|_{\text{F}},$$

will yield the desired lower bound of Theorem 1 on the minimax estimation error (Eq. (7)). To that end, we need to show the existence of a sufficiently large set $\mathcal{X}_\epsilon \subseteq \mathcal{X}(G)$ such that *(i)* the points in $\mathcal{X}_\epsilon$ are well separated under $d(\cdot,\cdot)$, while *(ii)* the KL divergence of the induced probability measures is upper appropriately bounded.

Lemma 2.2. (Local Packing) *Consider a (p,k,d)-layer graph G on p vertices with $k \geq 4$ and $\log d \geq 4 \cdot H(3/4)$. For any $\epsilon \in (0,1]$, there exists a set $\mathcal{X}_\epsilon \subset \mathcal{X}(G)$ such that*

$$\epsilon/\sqrt{2}\ <\ \|\mathbf{x}_i - \mathbf{x}_j\|_2\ \leq\ \sqrt{2} \cdot \epsilon,$$

for all $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}_\epsilon$, $\mathbf{x}_i \neq \mathbf{x}_j$, and

$$\log|\mathcal{X}_\epsilon|\ \geq\ \log\frac{p-2}{k} + {1}/{4} \cdot k \cdot \log d.$$

[1] A pseudometric on a set $\mathcal{X}$ is a function $d : \mathcal{Q}^2 \to \mathbb{R}$ that satisfies all properties of a distance (non-negativity, symmetry, triangle inequality) except the identity of indiscernibles: $d(\mathbf{q},\mathbf{q}) = 0$, $\forall \mathbf{q} \in \mathcal{Q}$ but possibly $d(\mathbf{q}_1, \mathbf{q}_2) = 0$ for some $\mathbf{q}_1 \neq \mathbf{q}_2 \in \mathcal{Q}$.

Proof. (See Appendix 7). □

For a set $\mathcal{X}_\epsilon$ with the properties of Lemma 2.2, taking into account the fact that $\|\mathbf{x}_i\mathbf{x}_i^\top - \mathbf{x}_j\mathbf{x}_j^\top\|_{\mathrm{F}} \geq \|\mathbf{x}_i - \mathbf{x}_j\|_2$ (Lemma A.1.2 of (Vu & Lei, 2012)), we have

$$d^2(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i\mathbf{x}_i^\top - \mathbf{x}_j\mathbf{x}_j^\top\|_{\mathrm{F}}^2 > \frac{\epsilon^2}{2} \triangleq \alpha^2. \quad (9)$$

$\forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}_\epsilon, \mathbf{x}_i \neq \mathbf{x}_j$. Moreover,

$$\begin{aligned}\mathrm{KL}(\mathcal{D}_p(\mathbf{x}_i) \,\|\, \mathcal{D}_p(\mathbf{x}_j)) &= \tfrac{\beta}{2(1+\beta)} \cdot \Big[(1+\beta)\times \\ &\mathrm{Tr}\Big(\big(\mathbf{I} - \mathbf{x}_j\mathbf{x}_j^\top\big)\mathbf{x}_i\mathbf{x}_i^\top\Big) - \mathrm{Tr}\Big(\mathbf{x}_j\mathbf{x}_j^\top\big(\mathbf{I} - \mathbf{x}_i\mathbf{x}_i^\top\big)\Big)\Big] \\ &= \tfrac{\beta^2}{4(1+\beta)} \cdot \|\mathbf{x}_i\mathbf{x}_i^\top - \mathbf{x}_j\mathbf{x}_j^\top\|_{\mathrm{F}}^2 \;\leq\; \tfrac{\beta^2}{(1+\beta)} \cdot \|\mathbf{x}_i - \mathbf{x}_j\|_2^2.\end{aligned}$$

In turn, for the n-fold product distribution, and taking into account that $\|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \sqrt{2} \cdot \epsilon$,

$$\mathrm{KL}\big(\mathcal{D}_p^{(n)}(\mathbf{x}_i) \,\|\, \mathcal{D}_p^{(n)}(\mathbf{x}_j)\big) \;\leq\; \frac{2n\beta^2\epsilon^2}{(1+\beta)} \;\triangleq\; \gamma. \quad (10)$$

Eq. (9) and (10) establish the parameters α and γ required by Lemma 2.1. Substituting those into (8), along with the lower bound of Lemma 2.2 on $|\mathcal{X}_\epsilon|$, we obtain

$$\max_{\mathbf{x}_i \in \mathcal{X}_\epsilon} \mathbb{E}_{\mathcal{D}_p^{(n)}(\mathbf{x}_i)}[d(\widehat{\mathbf{x}}, \mathbf{x}_i)] \geq \frac{\epsilon}{2\sqrt{2}}\left[1 - \frac{n\frac{2\epsilon^2\beta^2}{(1+\beta)} + \log 2}{\log|\mathcal{X}_\epsilon|}\right]. \quad (11)$$

The final step towards establishing the desired lower bound in (7) is to appropriately choose ϵ. One can verify that if

$$\epsilon^2 = \min\Big\{1,\ \tfrac{C'\cdot(1+\beta)}{\beta^2} \cdot \tfrac{1}{n}\left(\log\tfrac{p-2}{k} + \tfrac{k}{4}\cdot\log d\right)\Big\}, \quad (12)$$

where $C' > 0$ is a constant to be determined, then

$$n \cdot \frac{2\epsilon^2\beta^2}{(1+\beta)}\frac{1}{\log|\mathcal{X}_\epsilon|} \leq \frac{1}{4} \quad \text{and} \quad \log|\mathcal{X}_\epsilon| \geq 4\log 2, \quad (13)$$

(see Appendix 8 for details). Under the conditions in (13), the inequality in (11) implies that

$$\max_{\mathbf{x}_i \in \mathcal{X}_\epsilon} \mathbb{E}_{\mathcal{D}_p^{(n)}(\mathbf{x}_i)}[d(\widehat{\mathbf{x}}, \mathbf{x}_i)] \geq \tfrac{1}{2\sqrt{2}} \cdot \epsilon. \quad (14)$$

Substituting ϵ according to (12), yields the desired result in (7), completing the proof of Theorem 1.

2.2. Upper bound

Our upper bound is based on the estimator obtained via the constrained quadratic maximizaton in (3). We note that the analysis is not restricted to the spiked covariance model; it applies to a broader class of distributions (see Assum. 1).

Theorem 2 (Upper bound)**.** *Consider a (p, k, d)-layer graph G and $\mathbf{x}_\star \in \mathcal{X}(G)$. Let $\{\mathbf{y}_i\}_{i=1}^n$ be a sequence of n i.i.d. $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ samples, where $\mathbf{\Sigma} \succeq \mathbf{0}$ with eigenvalues $\lambda_1 > \lambda_2 \geq \ldots$, and principal eigenvector $\mathbf{x}_\star$. Let $\widehat{\mathbf{\Sigma}}$ be the empirical covariance of the n samples, $\widehat{\mathbf{x}}$ the estimate of $\mathbf{x}_\star$ obtained via* (3)*, and $\epsilon \triangleq \|\widehat{\mathbf{x}}\widehat{\mathbf{x}}^\top - \mathbf{x}_\star\mathbf{x}_\star^\top\|_{\mathrm{F}}$. Then,*

$$\mathbb{E}[\epsilon] \leq C \cdot \frac{\lambda_1}{\lambda_1 - \lambda_2} \cdot \frac{1}{n} \cdot \max\Big\{\sqrt{nA},\ A\Big\},$$

where $A = O\big(\log\frac{p-2}{k} + k\log d\big)$.

In the sequel, we provide a sketch proof of Theorem 2. The proof closely follows the steps of (Vu & Lei, 2012) in developing their upper bound for the the sparse PCA problem.

Lemma 2.3 (Lemma 3.2.1 (Vu & Lei, 2012))**.** *Consider $\mathbf{\Sigma} \in \mathbb{S}_+^{p\times p}$, with principal eigenvector $\mathbf{x}_\star$ and $\lambda_{\mathrm{gap}} \triangleq \lambda_1 - \lambda_2(\mathbf{\Sigma})$. For any $\widetilde{\mathbf{x}} \in \mathbb{R}^p$ with $\|\widetilde{\mathbf{x}}\|_2 = 1$,*

$$\tfrac{\lambda_1 - \lambda_2}{2} \cdot \|\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}^\top - \mathbf{x}_\star\mathbf{x}_\star^\top\|_{\mathrm{F}}^2 \;\leq\; \big\langle \mathbf{\Sigma},\, \mathbf{x}_\star\mathbf{x}_\star^\top - \widetilde{\mathbf{x}}\widetilde{\mathbf{x}}^\top \big\rangle.$$

Let $\widehat{\mathbf{x}}$ be an estimate of $\mathbf{x}_\star$ via (3), and $\epsilon \triangleq \|\widehat{\mathbf{x}}\widehat{\mathbf{x}}^\top - \mathbf{x}_\star\mathbf{x}_\star^\top\|_{\mathrm{F}}$. From Lemma 2.3, it follows (see (Vu & Lei, 2012)) that

$$\tfrac{\lambda_1 - \lambda_2}{2} \cdot \epsilon^2 \leq \big\langle \widehat{\mathbf{\Sigma}} - \mathbf{\Sigma},\, \widehat{\mathbf{x}}\widehat{\mathbf{x}}^\top - \mathbf{x}_\star\mathbf{x}_\star^\top \big\rangle. \quad (15)$$

Both $\mathbf{x}_\star$ and $\widehat{\mathbf{x}}$ belong to $\mathcal{X}(G)$: unit-norm vectors, with support of cardinality $k+2$ coinciding with a path in $\mathcal{P}(G)$. Their difference is supported in $\mathcal{P}^2(G)$: the collection of sets formed by the union of two sets in $\mathcal{P}(G)$. Let $\mathcal{X}^2(G)$ denote the set of unit norm vectors supported in $\mathcal{P}^2(G)$. Via an appropriate upper bounding of the right-hand side of (15), (Vu & Lei, 2012) show that

$$\mathbb{E}[\epsilon] \leq \tfrac{\widehat{C}}{\lambda_1 - \lambda_2} \cdot \mathbb{E}\Big[\sup_{\boldsymbol{\theta} \in \mathcal{X}^2} \big|\boldsymbol{\theta}^\top\big(\widehat{\mathbf{\Sigma}} - \mathbf{\Sigma}\big)\boldsymbol{\theta}\big|\Big],$$

for an appropriate constant $\widehat{C} > 0$. Further, under the assumptions on the data distribution, and utilizing a result due to (Mendelson, 2010),

$$\mathbb{E}\left[\sup_{\boldsymbol{\theta} \in \mathcal{X}^2} \big|\boldsymbol{\theta}^\top\big(\widehat{\mathbf{\Sigma}} - \mathbf{\Sigma}\big)\boldsymbol{\theta}\big|\right] \leq C'K^2\frac{\lambda_1}{n}\max\big\{\sqrt{n}A,\ A^2\big\},$$

for C' and K constants depending on the distribution, and

$$A \triangleq \mathbb{E}_{\mathbf{Y}\sim N(\mathbf{0}, \mathbf{I}_p)}\big[\sup_{\boldsymbol{\theta} \in \mathcal{X}^2}\langle \mathbf{Y}, \boldsymbol{\theta}\rangle\big]. \quad (16)$$

This reduces the problem of bounding $\mathbb{E}[\epsilon]$ to bounding the supremum of a Gaussian process. Let $\mathcal{N}_\delta \subset \mathcal{X}^2(G)$ be a minimal δ-covering of $\mathcal{X}^2(G)$ in the Euclidean metric with the property that $\forall \mathbf{x} \in \mathcal{X}^2(G)$, $\exists \mathbf{y} \in \mathcal{N}_\delta$ such that $\|\mathbf{x} - \mathbf{y}\|_2 \leq \delta$ and $\mathrm{supp}(\mathbf{x} - \mathbf{y}) \in \mathcal{P}^2(G)$. Then,

$$\sup_{\boldsymbol{\theta} \in \mathcal{X}^2}\langle \mathbf{Y}, \boldsymbol{\theta}\rangle \;\leq\; (1-\delta)^{-1} \cdot \max_{\boldsymbol{\theta} \in \mathcal{N}_\delta}\langle \mathbf{Y}, \boldsymbol{\theta}\rangle. \quad (17)$$

Taking expectation w.r.t. $\mathbf{Y}$ and applying a union bound on the right hand side, we conclude

$$A \leq \widetilde{C} \cdot (1-\delta)^{-1} \cdot \sqrt{\log|\mathcal{N}_\delta|}. \quad (18)$$

It remains to construct a δ-covering $\mathcal{N}_\delta$ with the desired properties. To this end, we associate isometric copies of $\mathbb{S}_2^{2k+1}$ with each support set in $\mathcal{P}^2(G)$. It is known that there exists a minimal δ-covering for $\mathbb{S}_2^{2k+1}$ with cardinality at most $(1 + {}^2/\delta)^{2k+2}$. The union of the local δ-nets forms a set $\mathcal{N}_\delta$ with the desired properties. Then,

$$\begin{aligned} \log |\mathcal{N}_\delta| \leq\ & \log |\mathcal{P}^2(G)| + 2(k+1)\log(1 + {}^2/\delta) \\ =\ & O\big(\log \tfrac{p-2}{k} + k \log d\big)\,, \end{aligned}$$

for any constant δ. Substituting in (18), implies the desired bound on $\mathbb{E}[\epsilon]$, completing the proof of Theorem 2.

3. Algorithmic approaches

We propose two algorithms for approximating the solution of the constrained quadratic maximization in (3):

1. The first is an adaptation of the *truncated power iteration* method of (Yuan & Zhang, 2013) for the problem of computing sparse eigenvectors.
2. The second relies on approximately solving (3) on a low rank approximation of $\widehat{\mathbf{\Sigma}}$, similar to (Papailiopoulos et al., 2013; Asteris et al., 2014).

Both algorithms rely on a projection operation from $\mathbb{R}^p$ onto the feasible set $\mathcal{X}(G)$, for a given graph $G = (V, E)$. Besides the projection step, the algorithms are oblivious to the specifics of the constraint set,[2] and can adapt to different constraints by modifying the projection operation.

3.1. Graph-Truncated Power Method

Algorithm 1 Graph-Truncated Power Method

input $\widehat{\mathbf{\Sigma}} \in \mathbb{R}^{p\times p},\ G = (V, E),\ \mathbf{x}_0 \in \mathbb{R}^p$
$i \leftarrow 0$
repeat
  $\mathbf{w}_i \leftarrow \widehat{\mathbf{\Sigma}}\mathbf{x}_i$
  $\mathbf{x}_{i+1} \leftarrow \text{Proj}_{\mathcal{X}(G)}(\mathbf{w}_i)$
  $i \leftarrow i + 1$
until Convergence/Stop Criterion
output $\mathbf{x}_i$

We consider a simple iterative procedure, similar to the truncated power method of (Yuan & Zhang, 2013) for the problem of computing sparse eigenvectors. Our algorithm produces sequence of vectors $\mathbf{x}_i \in \mathcal{X}(G)$, $i \geq 0$, that serve as intermediate estimates of the desired solution of (3).

The procedure is summarized in Algorithm 1. In the ith iteration, the current estimate $\mathbf{x}_i$ is multiplied by the empirical covariance $\widehat{\mathbf{\Sigma}}$, The product $\mathbf{w}_i \in \mathbb{R}^p$ is projected back to the feasible set $\mathcal{X}(G)$, yielding the next estimate $\mathbf{x}_{i+1}$.

[2]For Alg. 2, the observation holds under mild assumptions: $\mathcal{X}(G)$ must be such that $\|\mathbf{x}\|_2 = \Theta(1)$, while $\pm\mathbf{x} \in \mathcal{X}(G)$ should both achieve the same objective value.

The core of Algorithm 1 lies in the projection operation,

$$\text{Proj}_{\mathcal{X}(G)}(\mathbf{w}) \triangleq \underset{\mathbf{x}\in\mathcal{X}(G)}{\arg\min}\ \frac{1}{2}\|\mathbf{x} - \mathbf{w}\|_2^2, \tag{19}$$

which is analyzed separately in Section 3.3. The initial estimate $\mathbf{x}_0$ can be selected randomly or based on simple heuristics, *e.g.*, the projection on $\mathcal{X}(G)$ of the column of $\widehat{\mathbf{\Sigma}}$ corresponding to the largest diagonal entry. The algorithm terminates when some convergence criterion is satisfied.

The computational complexity (per iteration) of Algorithm 1 is dominated by the cost of matrix-vector multiplication and the projection step. The former is $O(k \cdot p)$, where k is cardinality of the largest support in $\mathcal{X}(G)$. The projection operation for the particular set $\mathcal{X}(G)$, boils down to solving the longest path problem on a weighted variant of the DAG G (see Section 3.3), which can be solved in time $O(|V| + |E|)$, *i.e.*, linear in the size of G.

3.2. Low-Dimensional Sample and Project

The second algorithm outputs an estimate of the desired solution of (3) by (approximately) solving the constrained quadratic maximization *not* on the original matrix $\widehat{\mathbf{\Sigma}}$, but on a low rank approximation $\widehat{\mathbf{\Sigma}}_r$ of $\widehat{\mathbf{\Sigma}}$, instead:

$$\widehat{\mathbf{\Sigma}}_r = \sum_{i=1}^{r} \lambda_i \mathbf{q}_i \mathbf{q}_i^\top = \sum_{i=1}^{r} \mathbf{v}_i \mathbf{v}_i^\top = \mathbf{V}\mathbf{V}^\top, \tag{20}$$

where λ_i is the ith largest eigenvalue of $\widehat{\mathbf{\Sigma}}$, $\mathbf{q}_i$ is the corresponding eigenvector, $\mathbf{v}_i \triangleq \sqrt{\lambda_i} \cdot \mathbf{q}_i$, and $\mathbf{V}$ is the $p \times r$ matrix whose ith column is equal to $\mathbf{v}_i$. The approximation rank r is an accuracy parameter; typically, $r \ll p$.

Our algorithm operates[3] on $\widehat{\mathbf{\Sigma}}_r$ and seeks

$$\mathbf{x}_r \triangleq \underset{\mathbf{x}\in\mathcal{X}(G)}{\arg\max}\ \mathbf{x}^\top \widehat{\mathbf{\Sigma}}_r \mathbf{x}. \tag{21}$$

The motivation is that an (approximate) solution for the low-rank problem in (21) can be efficiently computed. Intuitively, if $\widehat{\mathbf{\Sigma}}_r$ is a sufficiently good approximation of the original matrix $\widehat{\mathbf{\Sigma}}$, then $\mathbf{x}_r$ would perform similarly to the solution $\mathbf{x}_\star$ of the original problem (3).

The Algorithm. Our algorithm samples points from the low-dimensional principal subspace of $\widehat{\mathbf{\Sigma}}$, and projects them on the feasible set $\mathcal{X}(G)$, producing a set of candidate estimates for $\mathbf{x}_r$. It outputs the candidate that maximizes the objective in (21). The exact steps are formally presented in Algorithm 2. The following paragraphs delve into the details of Algorithm 2.

[3]Under the spiked covariance model, this approach may be asymptotically unsuitable; as the ambient dimension increases, it with fail to recover the latent signal. Empirically, however, if the spectral decay of $\widehat{\mathbf{\Sigma}}$ is sharp, it yields very competitive results.

Algorithm 2 Low-Dimensional Sample and Project

input $\widehat{\mathbf{\Sigma}} \in \mathbb{R}^{p \times p}$, $G = (V, E)$, $r \in [p]$, $\epsilon > 0$

$[\mathbf{Q}, \mathbf{\Lambda}] \leftarrow \mathtt{svd}(\widehat{\mathbf{\Sigma}}, r)$

$\mathbf{V} \leftarrow \mathbf{Q}\mathbf{\Lambda}^{1/2}$ $\quad \{\widehat{\mathbf{\Sigma}}_r \triangleq \mathbf{V}\mathbf{V}^\top\}$

$\mathcal{C} \leftarrow \emptyset$ $\quad$ {Candidate solutions}

for $i = 1 : O(\epsilon^{-r} \cdot \log p)$ **do**

  $\mathbf{c}_i \leftarrow$ uniformly sampled from $\mathbb{S}^{r-1}$

  $\mathbf{w}_i \leftarrow \mathbf{V}\mathbf{c}_i$

  $\mathbf{x}_i \leftarrow \text{Proj}_{\mathcal{X}(G)}(\mathbf{w}_i)$

  $\mathcal{C} = \mathcal{C} \cup \{\mathbf{x}_i\}$

end for

output $\widehat{\mathbf{x}}_r \leftarrow \arg\max_{\mathbf{x} \in \mathcal{C}} \|\mathbf{V}^\top \mathbf{x}\|_2^2$

3.2.1. The Low Rank Problem

The rank-r maximization in (21) can be written as

$$\max_{\mathbf{x} \in \mathcal{X}(G)} \mathbf{x}^\top \widehat{\mathbf{\Sigma}}_r \mathbf{x} = \max_{\mathbf{x} \in \mathcal{X}(G)} \|\mathbf{V}^\top \mathbf{x}\|_2^2, \tag{22}$$

and in turn (see (Asteris et al., 2014) for details), as a double maximization over the variables $\mathbf{c} \in \mathbb{S}^{r-1}$ and $\mathbf{x} \in \mathbb{R}^p$:

$$\max_{\mathbf{x} \in \mathcal{X}(G)} \|\mathbf{V}^\top \mathbf{x}\|_2^2 = \max_{\mathbf{c} \in \mathbb{S}^{r-1}} \max_{\mathbf{x} \in \mathcal{X}(G)} \left((\mathbf{V}\mathbf{c})^\top \mathbf{x} \right)^2. \tag{23}$$

The rank-1 case. Let $\mathbf{w} \triangleq \mathbf{V}\mathbf{c}$; $\mathbf{w}$ is only a vector in $\mathbb{R}^p$. For given $\mathbf{c}$ and $\mathbf{w}$, the $\mathbf{x}$ that maximizes the objective in (23) (as a function of $\mathbf{c}$) is

$$\mathbf{x}(\mathbf{c}) \in \arg\max_{\mathbf{x} \in \mathcal{X}(G)} \left(\mathbf{w}^\top \mathbf{x} \right)^2. \tag{24}$$

The maximization in (24) is nothing but a rank-1 instance of the maximization in (22). Observe that if $\mathbf{x} \in \mathcal{X}(G)$, then $-\mathbf{x} \in \mathcal{X}(G)$, and the two vectors attain the same objective value. Hence, (24) can be simplified:

$$\mathbf{x}(\mathbf{c}) \in \arg\max_{\mathbf{x} \in \mathcal{X}(G)} \mathbf{w}^\top \mathbf{x}. \tag{25}$$

Further, since $\|\mathbf{x}\|_2 = 1$, $\forall \mathbf{x} \in \mathcal{X}(G)$, the maximization in (25) is equivalent to minimizing $\frac{1}{2}\|\mathbf{w} - \mathbf{x}\|_2^2$. In other words, $\mathbf{x}(\mathbf{c})$ is just the projection of $\mathbf{w} \in \mathbb{R}^p$ onto $\mathcal{X}(G)$:

$$\mathbf{x}(\mathbf{c}) \in \text{Proj}_{\mathcal{X}(G)}(\mathbf{w}). \tag{26}$$

The projection operator is described in Section 3.3.

Multiple rank-1 instances. Let $(\mathbf{c}_r, \mathbf{x}_r)$ denote a pair that attains the maximum value in (23). If $\mathbf{c}_r$ was known, then $\mathbf{x}_r$ would coincide with the projection $\mathbf{x}(\mathbf{c}_r)$ of $\mathbf{w} = \mathbf{V}\mathbf{c}_r$ on the feasible set, according to (26).

Of course, the optimal value $\mathbf{c}_r$ of the auxiliary variable is not known. Recall, however, that $\mathbf{c}_r$ lies on the low dimensional manifold $\mathbb{S}^{r-1}$. Consider an ϵ-net $\mathcal{N}_\epsilon$ covering the r-dimensional unit sphere $\mathbb{S}^{r-1}$; Algorithm 2 constructs such a net by random sampling. By definition, $\mathcal{N}_\epsilon$ contains at least one point, call it $\widehat{\mathbf{c}}_r$, in the vicinity of $\mathbf{c}_r$. It can be shown that the corresponding solution $\mathbf{x}(\widehat{\mathbf{c}}_r)$ in (24) will perform approximately as well as the optimal solution $\mathbf{x}_r$, in terms of the quadratic objective in (23), for a large, but tractable, number of points in the ϵ-net of $\mathbb{S}^{r-1}$.

3.3. The Projection Operator

Algorithms 1, and 2 rely on a projection operation from $\mathbb{R}^p$ onto the feasible set $\mathcal{X}(G)$ (Eq. (4)). We show that the projection effectively reduces to solving the *longest path problem* on (a weighted variant of) G.

The projection operation, defined in Eq. (19), can be equivalently[4] written as

$$\text{Proj}_{\mathcal{X}(G)}(\mathbf{w}) \triangleq \arg\max_{\mathbf{x} \in \mathcal{X}(G)} \mathbf{w}^\top \mathbf{x}.$$

For any $\mathbf{x} \in \mathcal{X}(G)$, $\text{supp}(\mathbf{x}) \in \mathcal{P}(G)$. For a given set π, by the Cauchy-Schwarz inequality,

$$\mathbf{w}^\top \mathbf{x} = \textstyle\sum_{i \in \pi} w_i x_i \;\leq\; \sum_{i \in \pi} w_i^2 = \widehat{\mathbf{w}}^\top \mathbf{1}_\pi, \tag{27}$$

where $\widehat{\mathbf{w}} \in \mathbb{R}^p$ is the vector obtained by squaring the entries of $\mathbf{w}$, *i.e.*, $\widehat{w}_i = w_i^2$, $\forall i \in [n]$, and $\mathbf{1}_\pi \in \{0,1\}^p$ denotes the characteristic of π. Letting $\mathbf{x}[\pi]$ denote the subvector of $\mathbf{x}$ supported on π, equality in (27) can be achieved by $\mathbf{x}$ such that $\mathbf{x}[\pi] = \mathbf{w}[\pi]/\|\mathbf{w}[\pi]\|_2$, and $\mathbf{x}[\pi^c] = \mathbf{0}$.

Hence, the problem in (27) reduces to determining

$$\pi(\mathbf{w}) \in \arg\max_{\pi \in \mathcal{P}(G)} \widehat{\mathbf{w}}^\top \mathbf{1}_\pi. \tag{28}$$

Consider a weighted graph $G_{\mathbf{w}}$, obtained from $G = (V, E)$ by assigning weight $\widehat{w}_v = w_v^2$ on vertex $v \in V$. The objective function in (28) equals the *weight of the path* π in $G_{\mathbf{w}}$, *i.e.*, the sum of weights of the vertices along π. Determining the optimal support $\pi(\mathbf{w})$ for a given $\mathbf{w}$, is equivalent to solving the *longest (weighted) path problem*[5] on $G_{\mathbf{w}}$.

The longest (weighted) path problem is NP-hard on arbitrary graphs. In the case of DAGs, however, it can be solved using standard algorithms relying on topological sorting in time $O(|V| + |E|)$ (Cormen et al., 2001), *i.e.*, linear in the size of the graph. Hence, the projection $\mathbf{x}$ can be determined in time $O(p + |E|)$.

---

[4] It follows from expanding the quadratic $\frac{1}{2}\|\mathbf{x} - \mathbf{w}\|_2^2$ and the fact that $\|\mathbf{x}\|_2 = 1, \forall \mathbf{x} \in \mathcal{X}(G)$.

[5] The longest path problem is commonly defined on graphs with weighted edges instead of vertices. The latter is trivially transformed to the former: set $w(u, v) \leftarrow w(v)$, $\forall (u, v) \in E$, where $w(u, v)$ denotes the weight of edge (u, v), and $w(v)$ that of vertex v. Auxiliary edges can be introduced for source vertices.

4. Experiments

4.1. Synthetic Data.

We evaluate Alg. 1 and 2 on synthetic data, generated according to the model of Sec. 2. We consider two metrics: the loss function $\|\widehat{\mathbf{x}}\widehat{\mathbf{x}}^\top - \mathbf{x}_\star \mathbf{x}_\star\|_\text{F}$ and the Support Jaccard distance between the true signal $\mathbf{x}_\star$ and the estimate $\widehat{\mathbf{x}}$.

For dimension p, we generate a (p, k, d)-layer graph G, with $k = \log p$ and out-degree $d = p/k$, *i.e.*, each vertex is connected to all vertices in the following layer. We augment the graph with auxiliary source and terminal vertices S and T with edges to the original vertices as in Fig. 1.

Per random realization, we first construct a signal $\mathbf{x}_\star \in \mathcal{X}(G)$ as follows: we randomly select an S-T path π in G, and assign random zero-mean Gaussian values to the entries of $\mathbf{x}_\star$ indexed by π. The signal is scaled to unit length. Given $\mathbf{x}_\star$, we generate n independent samples according to the spiked covariance model in (5).

Fig. 2 depicts the aforementioned distance metrics as a function of the number n of observations. Results are the average of 100 independent realizations. We repeat the procedure for multiple values of the ambient dimension p.

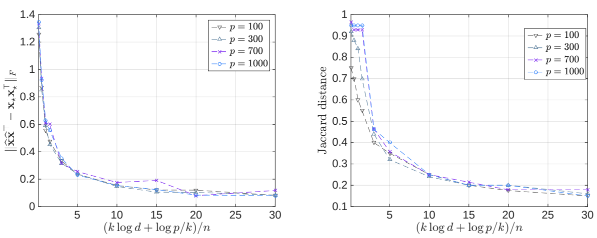

Figure 2. Metrics on the estimate $\widehat{\mathbf{x}}$ produced by Alg. 1 (Alg. 2 is similar) as a function of the sample number (average of 100 realizations). Samples are generated according to the spiked covariance model with signal $\mathbf{x}_\star \in \mathcal{X}(G)$ for a (p, k, d)-layer graph G. Here, $k = \log p$ and $d = p/k$. We repeat for multiple values of p.

Comparison with Sparse PCA. We compare the performance of Alg. 1 and Alg. 2 with their sparse PCA counterparts: the Truncated Power Method of (Yuan & Zhang, 2013) and the Spannogram Alg. of (Papailiopoulos et al., 2013), respectively.

Fig. 3 depicts the metrics of interest as a function of the number of samples, for all four algorithms. Here, samples are drawn *i.i.d* from $N(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ has principal eigenvector equal to $\mathbf{x}_\star$, and power law spectral decay: $\lambda_i = i^{-1/4}$. Results are an average of 100 realizations.

The side information on the structure of $\mathbf{x}_\star$ assists the recovery: both algorithms achieve improved performance compared to their sparse PCA counterparts. Here, the power method based algorithms exhibit inferior performance, which may be attributed to poor initialization. We note, though, that at least for the size of these experiments, the power method algorithms are significantly faster.

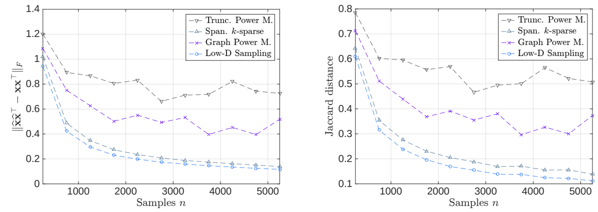

Figure 3. Estimation error between true signal $\mathbf{x}_\star$ and estimate $\widehat{\mathbf{x}}$ from n samples. (average of 100 realizations). Samples generated *i.i.d.* $\sim N(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ has eigenvalues $\lambda_i = i^{-1/4}$ and principal eigenvector $\mathbf{x}_\star \in \mathcal{X}(G)$, for a (p, k, d)-layer graph G. ($p = 10^3$, $k = 50$, $d = 10$).

4.2. Finance Data.

This dataset contains daily closing prices for 425 stocks of the S&P 500 Index, over a period of 1259 days (5-years): 02.01.2010 – 01.28.2015, collected from Yahoo! Finance[6]. Stocks are classified, according to the *Global Industry Classification Standard*[7] (GICS), into 10 business *sectors e.g.*, Energy, Health Care, Information Technology, etc (see Fig. 4 for the complete list).

We seek a set of stocks comprising a single representative from each GICS sector, which captures most of the variance in the dataset. Equivalently, we want to compute a structured principal component constrained to have exactly 10 nonzero entries; one for each GICS sector.

Consider a layer graph $G = (V, E)$ (similar to the one depicted in Fig. 1) on $p = 425$ vertices corresponding to the 425 stocks, partitioned into $k = 10$ groups (layers) $\mathcal{L}_1, \ldots, \mathcal{L}_{10} \subseteq V$, corresponding to the GICS sectors. Each vertex in layer $\mathcal{L}_i$ has outgoing edges towards all (and only the) vertices in layer $\mathcal{L}_{i+1}$. Note that (unlike Fig. 1) layers do *not* have equal sizes, and the vertex out-degree varies across layers. Finally, we introduce auxiliary vertices S and T connected with the original graph as in Fig. 1.

Observe that any set of sector-representatives corresponds to an S-T path in G, and vice versa. Hence, the desired set of stocks can be obtained by finding a *structured* principal component constrained to be supported along an S-T path in G. Note that the order of layers in G is irrelevant.

Fig. 4 depicts the subset of stocks selected by the proposed structure PCA algorithms (Alg. 1, 2). A single representative is selected from each sector. For comparison, we also run two corresponding algorithms for sparse PCA, with

[6] `http://finance.yahoo.com`

[7] `http://www.msci.com/products/indexes/sector/gics/gics_structure.html`

sparsity parameter $k = 10$, equal to the number of sectors. As expected, the latter yield components achieving higher values of explained variance, but the selected stocks originate from only 5 out of the 10 sectors.

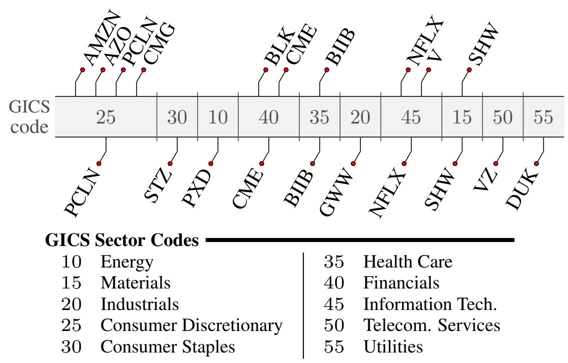

Figure 4. The figure depicts the sets of 10 stocks extracted by sparse PCA and our structure PCA approach. Sparse PCA ($k = 10$), selects 10 stocks from 5 GICS sectors (above). On the contrary, our structured PCA algorithms yield a set of 10 stocks containing a representative from each sector (below) as desired.

4.3. Neuroscience Data.

We use a single-session/single-participant resting state functional magnetic resonance imaging (resting state fMRI) dataset. The participant was not instructed to perform any explicit cognitive task throughout the scan (Van Essen et al., 2013). Data was provided by the Human Connectome Project, WU-Minn Consortium.[8]

Mean timeseries of $n = 1200$ points for $p = 111$ regions-of-interest (ROIs) are extracted based on the Harvard-Oxford Atlas (Desikan et al., 2006). The timescale of analysis is restricted to 0.01–0.1Hz. Based on recent results on resting state fMRI neural networks, we set the posterior cingulate cortex as a source node S, and the prefrontal cortex as a target node T (Greicius et al., 2009). Starting from S, we construct a layered graph with $k = 4$, based on the physical (Euclidean) distances between the center of mass of the ROIs: *i.e.,* given layer $\mathcal{L}_i$, we construct $\mathcal{L}_{i+1}$ from non-selected nodes that are close in the Euclidean sense. Here, $|\mathcal{L}_1| = 34$ and $|\mathcal{L}_i| = 25$ for $i = 2, 3, 4$. Each layer is fully connected with its previous one. No further assumptions are derived from neurobiology.

The extracted component suggests a directed pathway from the posterior cingulate cortex (S) to the prefrontal cortex (T), through the hippocampus (1), nucleus accumbens (2), parahippocampal gyrus (3), and frontal operculum (4) (Fig. 5). Hippocampus and the parahippocampal gyrus are critical in memory encoding, and have been found to be structurally connected to the posterior cingulate cortex and the prefrontal cortex (Greicius et al., 2009). The nucleus accumbens receives input from the hippocampus, and plays an important role in memory consolidation (Wittmann et al., 2005). It is noteworthy that our approach has pinpointed the core neural components of the memory network, given minimal information.

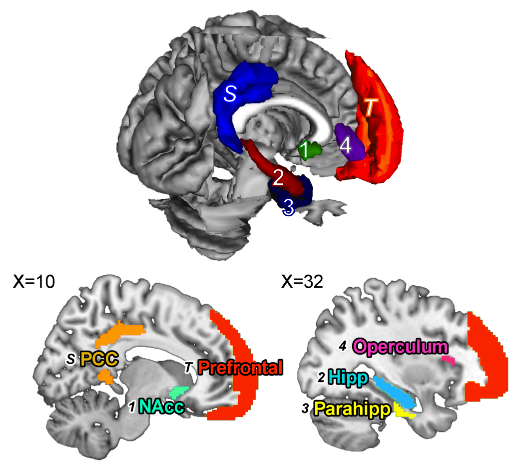

Figure 5. We highlight the nodes extracted for the neuroscience example. Source node set to the posterior cingulate cortex (S: PCC), and target to the prefrontal cortex (T: Prefrontal). The directed path proceeded from the nucleus accumbens (1: NAcc), hippocampus (2: Hipp), parahippocampal gyrus (3: Parahipp), and to the frontal operculum (4: Operculum). Here, X coordinates (in mm) denote how far from the midline the cuts are.

5. Conclusions

We introduced a new problem: sparse PCA where the set of feasible support sets is determined by a graph on the variables. We focused on the special case where feasible sparsity patterns coincide with paths on the underlying graph. We provided an upper bound on the statistical complexity of the constrained quadratic maximization estimator (3), under a simple graph model, complemented with a lower bound on the minimax error. Finally, we proposed two algorithms to extract a component accommodating the graph constraints and applied them on real data from finance and neuroscience.

A potential future direction is to expand the set of graph-induced sparsity patterns (beyond paths) that can lead to interpretable solutions and are computationally tractable. We hope this work triggers future efforts to introduce and exploit such underlying structure in diverse research fields.

[8](Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54MH091657) funded by the 16 NIH Institutes and Centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University.

6. Acknowledgments

The authors would like to acknowledge support from grants: NSF CCF 1422549, 1344364, 1344179 and an ARO YIP award.

References

Amini, Arash and Wainwright, Martin. High-dimensional analysis of semidefinite relaxations for sparse principal components. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, pp. 2454–2458. IEEE, 2008.

Amini, Arash and Wainwright, Martin. High-dimensional analysis of semidefinite relaxations for sparse principal components. *The Annals of Statistics*, pp. 2877–2921, 2009.

Asteris, Megasthenis, Papailiopoulos, Dimitris, and Dimakis, Alexandros. Nonnegative sparse PCA with provable guarantees. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1728–1736, 2014.

Baldassarre, Luca, Bhan, Nirav, Cevher, Volkan, and Kyrillidis, Anastasios. Group-sparse model selection: Hardness and relaxations. *arXiv preprint arXiv:1303.3207*, 2013.

Baraniuk, R., Cevher, V., Duarte, M., and Hegde, C. Model-based compressive sensing. *Information Theory, IEEE Transactions on*, 56(4):1982–2001, 2010.

Cormen, Thomas, Stein, Clifford, Rivest, Ronald, and Leiserson, Charles. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition, 2001. ISBN 0070131511.

d'Aspremont, Alexandre, El Ghaoui, Laurent, Jordan, Michael, and Lanckriet, Gert. A direct formulation for sparse PCA using semidefinite programming. *SIAM review*, 49(3):434–448, 2007.

d'Aspremont, Alexandre, Bach, Francis, and Ghaoui, Laurent El. Optimal solutions for sparse principal component analysis. *The Journal of Machine Learning Research*, 9:1269–1294, 2008.

Desikan, Rahul, Ségonne, Florent, Fischl, Bruce, Quinn, Brian, Dickerson, Bradford, Blacker, Deborah, Buckner, Randy, Dale, Anders, Maguire, Paul, and Hyman, Bradley. An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. *Neuroimage*, 31(3):968–980, 2006.

Greicius, Michael D, Supekar, Kaustubh, Menon, Vinod, and Dougherty, Robert F. Resting-state functional connectivity reflects structural connectivity in the default mode network. *Cerebral cortex*, 19(1):72–78, 2009.

Jenatton, Rodolphe, Obozinski, Guillaume, and Bach, Francis. Structured sparse principal component analysis. In *International Conference on Artificial Intelligence and Statistics*, pp. 366–373, 2010.

Johnstone, Iain and Lu, Arthur Yu. Sparse principal components analysis. *Unpublished manuscript*, 2004.

Kyrillidis, Anastasios and Cevher, Volkan. Combinatorial selection and least absolute shrinkage via the CLASH algorithm. In *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, pp. 2216–2220. IEEE, 2012.

Mairal, Julien and Yu, Bin. Path coding penalties for directed acyclic graphs. In *Proceedings of the 4th NIPS Workshop on Optimization for Machine Learning (OPT11)*. Citeseer, 2011.

Mendelson, Shahar. Empirical processes with a bounded ψ-1 diameter. *Geometric and Functional Analysis*, 20(4):988–1027, 2010.

Papailiopoulos, Dimitris, Dimakis, Alex, and Korokythakis, Stavros. Sparse PCA through low-rank approximations. In *Proceedings of the 30th International Conference on Machine Learning*, ICML '13, pp. 767–774. ACM, 2013.

Van Essen, David, Smith, Stephen, Barch, Deanna, Behrens, Timothy, Yacoub, Essa, Ugurbil, Kamil, and Consortium, WU-Minn HCP. The WU-Minn human connectome project: An overview. *Neuroimage*, 80:62–79, 2013.

Vu, Vincent and Lei, Jing. Minimax rates of estimation for sparse PCA in high dimensions. In *International Conference on Artificial Intelligence and Statistics*, pp. 1278–1286, 2012.

Wittmann, Bianca, Schott, Björn, Guderian, Sebastian, Frey, Julietta, Heinze, Hans-Jochen, and Düzel, Emrah. Reward-related fMRI activation of dopaminergic midbrain is associated with enhanced hippocampus-dependent long-term memory formation. *Neuron*, 45(3):459–467, 2005.

Yu, Bin. Assouad, Fano, and Le Cam. In *Festschrift for Lucien Le Cam*, pp. 423–435. Springer, 1997.

Yuan, Xiao-Tong and Zhang, Tong. Truncated power method for sparse eigenvalue problems. *The Journal of Machine Learning Research*, 14(1):899–925, 2013.

Zou, Hui, Hastie, Trevor, and Tibshirani, Robert. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.

7. Proof of Lemma 2.2 – Local Packing Set

Towards the proof of Lemma 2.2, we develop a modified version of the Varshamov-Gilbert Lemma adapted to our specific model: the set of characteristic vectors of the S-T paths of a (p, k, d)-layer graph G.

Let $\delta_H(\mathbf{x}, \mathbf{y})$ denote the Hamming distance between two points $\mathbf{x}, \mathbf{y} \in \{0, 1\}^p$:

$$\delta_H(\mathbf{x}, \mathbf{y}) \triangleq |\{i : \; x_i \neq y_i\}| \, .$$

Lemma 7.4. *Consider a (p, k, d)-layer graph G on p vertices and the collection $\mathcal{P}(G)$ of S-T paths in G. Let*

$$\Omega \triangleq \big\{\mathbf{x} \in \{0, 1\}^p : \; \mathit{supp}(\mathbf{x}) \in \mathcal{P}(G)\big\},$$

i.e., the set of characteristic vectors of all S-T paths in G. For every $\xi \in (0, 1)$, there exists a set, $\Omega_\xi \subset \Omega$ such that

$$\delta_H(\mathbf{x}, \mathbf{y}) > 2(1 - \xi) \cdot k, \quad \forall \mathbf{x}, \mathbf{y} \in \Omega_\xi, \mathbf{x} \neq \mathbf{y}, \tag{29}$$

and

$$\log |\Omega_\xi| \geq \log \tfrac{p-2}{k} + (\xi \cdot k - 1) \cdot \log d - k \cdot H(\xi), \tag{30}$$

where $H(\cdot)$ is the binary entropy function.

Proof. Consider a labeling $1, \ldots, p$ of the p vertices in G, such that variable ω_i is associated with vertex i. Each point $\boldsymbol{\omega} \in \Omega$ is the characteristic vector of a set in $\mathcal{P}(G)$; nonzero entries of $\boldsymbol{\omega}$ correspond to vertices along an S-T path in G. With a slight abuse of notation, we refer to $\boldsymbol{\omega}$ as a path in G. Due to the structure of the (p, k, d)-layer graph G, all points in Ω have exactly $k + 2$ nonzero entries, *i.e.*,

$$\delta_H(\boldsymbol{\omega}, \mathbf{0}) = k + 2, \qquad \forall \boldsymbol{\omega} \in \Omega.$$

Each vertex in $\boldsymbol{\omega}$ lies in a distinct layer of G. In turn, for any pair of points $\boldsymbol{\omega}, \boldsymbol{\omega}' \in \Omega$,

$$\delta_H(\boldsymbol{\omega}, \boldsymbol{\omega}') = 2 \cdot \big(k - |\{i : \; \omega_i = \omega'_i = 1\}| - 2\big). \tag{31}$$

Note that the Hamming distance between the two points is a linear function of the number of their common nonzero entries, while it can take only even values with a maximum value of $2k$.

Without loss of generality, let S and T corresponding to vertices 1 and p, respectively. Then, the above imply that

$$\omega_1 = \omega_p = 1, \quad \forall \boldsymbol{\omega} \in \Omega.$$

Consider a fixed point $\widehat{\boldsymbol{\omega}} \in \Omega$, and let $\mathcal{B}(\widehat{\boldsymbol{\omega}}, r)$ denote the Hamming ball of radius r centered at $\widehat{\boldsymbol{\omega}}$, *i.e.*,

$$\mathcal{B}(\widehat{\boldsymbol{\omega}}, r) \;\triangleq\; \big\{\boldsymbol{\omega} \in \{0, 1\}^p : \; \delta_H(\widehat{\boldsymbol{\omega}}, \boldsymbol{\omega}) \leq r\big\}.$$

The intersection $\mathcal{B}(\widehat{\boldsymbol{\omega}}, r) \cap \Omega$ corresponds to S-T paths in G that have at least $k - r/2$ additional vertices in common with $\widehat{\boldsymbol{\omega}}$ besides vertices 1 and p that are common to all paths in Ω:

$$\begin{aligned} &\mathcal{B}(\widehat{\boldsymbol{\omega}}, r) \cap \Omega \\ &= \{\boldsymbol{\omega} \in \Omega : \delta_H(\widehat{\boldsymbol{\omega}}, \boldsymbol{\omega}) \leq r\} \\ &= \big\{\boldsymbol{\omega} \in \Omega : |\{i : \; \widehat{\omega}_i = \omega_i = 1\}| \geq k - \tfrac{r}{2} + 2\big\}, \end{aligned}$$

where the last equality is due to (31). In fact, due to the structure of G, the set $\mathcal{B}(\widehat{\boldsymbol{\omega}}, r) \cap \Omega$ corresponds to the S-T paths that *meet* $\widehat{\boldsymbol{\omega}}$ in at least $k - r/2$ intermediate layers. Taking into account that $|\Gamma_{\text{in}}(v)| = |\Gamma_{\text{out}}(v)| = d$, for all vertices v in $V(G)$ (except those in the first and last layer),

$$|\mathcal{B}(\widehat{\boldsymbol{\omega}}, r) \cap \Omega| \leq \binom{k}{k - \frac{r}{2}} \cdot d^{k - \left(k - \frac{r}{2}\right)} = \binom{k}{k - \frac{r}{2}} \cdot d^{\frac{r}{2}}.$$

Now, consider a *maximal* set $\Omega_\xi \subset \Omega$ satisfying (29), *i.e.*, a set that cannot be augmented by any other point in Ω. The union of balls $\mathcal{B}(\boldsymbol{\omega}, 2(1 - \xi) \cdot (k - 1))$ over all $\boldsymbol{\omega} \in \Omega_\xi$ covers Ω. To verify that, note that if there exists $\boldsymbol{\omega}' \in \Omega \backslash \Omega_\xi$ such that $\delta_H(\boldsymbol{\omega}, \boldsymbol{\omega}') > 2(1 - \xi) \cdot (k - 1)$, $\forall \boldsymbol{\omega} \in \Omega_\xi$, then $\Omega_\xi \cup \{\boldsymbol{\omega}'\}$ satisfies (29) contradicting the maximality of Ω_ξ. Based on the above,

$$\begin{aligned} |\Omega| &\leq \sum_{\boldsymbol{\omega} \in \Omega_\xi} |\mathcal{B}(\boldsymbol{\omega}, 2(1 - \xi) \cdot k) \cap \Omega| \\ &\leq \sum_{\mathbf{x} \in \Omega_\xi} \tbinom{k}{k - (1 - \xi)k} \cdot d^{(1 - \xi) \cdot k} \\ &\leq \sum_{\mathbf{x} \in \Omega_\xi} \tbinom{k}{\xi k} \cdot d^{(1 - \xi) \cdot k} \\ &\leq |\Omega_\xi| \cdot 2^{k \cdot H(\xi)} \cdot d^{(1 - \xi) \cdot k}. \end{aligned}$$

Taking into account that

$$|\Omega| = |\mathcal{P}(G)| = \frac{p - 2}{k} \cdot d^{k - 1},$$

we conclude that

$$\frac{p - 2}{k} \cdot d^{k - 1} \leq |\Omega_\xi| \cdot 2^{k \cdot H(\xi)} \cdot d^{(1 - \xi) \cdot k},$$

from which the desired result follows. □

Lemma 2.2. (Local Packing) *Consider a (p, k, d)-layer graph G on p vertices with $k \geq 4$ and $\log d \geq 4 \cdot H(3/4)$. For any $\epsilon \in (0, 1]$, there exists a set $\mathcal{X}_\epsilon \subset \mathcal{X}(G)$ such that*

$$\epsilon / \sqrt{2} \;<\; \|\mathbf{x}_i - \mathbf{x}_j\|_2 \;\leq\; \sqrt{2} \cdot \epsilon,$$

for all $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}_\epsilon$, $\mathbf{x}_i \neq \mathbf{x}_j$, and

$$\log |\mathcal{X}_\epsilon| \;\geq\; \log \frac{p - 2}{k} + \frac{1}{4} \cdot k \log d.$$

Proof. Without loss of generality, consider a labeling $1, \ldots, p$ of the p vertices in G, such that S and T correspond to vertices 1 and p, respectively. Let

$$\Omega \triangleq \{\mathbf{x} \in \{0,1\}^p : \operatorname{supp}(\mathbf{x}) \in \mathcal{P}(G)\},$$

where $\mathcal{P}(G)$ is the collection of S-T paths in G. By Lemma 7.4, and for $\xi = 3/4$, there exists a set $\Omega_\xi \subseteq \Omega$ such that

$$\delta_H(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j) > \frac{1}{2} \cdot k, \tag{32}$$

$\forall \boldsymbol{\omega}_i, \boldsymbol{\omega}_j \in \Omega_\xi$, $\boldsymbol{\omega}_i \neq \boldsymbol{\omega}_j$, and,

$$\begin{aligned}\log|\Omega_\xi| &\geq \log \tfrac{p-2}{k} + \left(\tfrac{3}{4} \cdot k - 1\right)\log d - k \cdot H\left(\tfrac{3}{4}\right) \\ &\geq \log \tfrac{p-2}{k} + \tfrac{2}{4} \cdot k \cdot \log d - k \cdot H\left(\tfrac{3}{4}\right) \\ &\geq \log \tfrac{p-2}{k} + \tfrac{1}{4} \cdot k \cdot \log d \end{aligned} \tag{33}$$

where the second and third inequalites hold under the assumptions of the lemma; $k \geq 4$ and $\log d \geq 4 \cdot H(3/4)$.

Consider the bijective mapping $\psi : \Omega_\xi \to \mathbb{R}^p$ defined as

$$\psi(\boldsymbol{\omega}) = \left[\sqrt{\tfrac{(1-\epsilon^2)}{2}} \cdot \omega_1,\ \tfrac{\epsilon}{\sqrt{k}} \cdot \boldsymbol{\omega}_{2:p-1},\ \sqrt{\tfrac{(1-\epsilon^2)}{2}} \cdot \omega_p\right].$$

We show that the set

$$\mathcal{X}_\epsilon \triangleq \{\psi(\boldsymbol{\omega}) :\ \boldsymbol{\omega} \in \Omega_\xi\}.$$

has the desired properties. First, to verify that $\mathcal{X}_\epsilon$ is a subset of $\mathcal{X}(G)$, note that $\forall \boldsymbol{\omega} \in \Omega_\xi \subset \Omega$,

$$\operatorname{supp}(\psi(\boldsymbol{\omega})) = \operatorname{supp}(\boldsymbol{\omega}) \in \mathcal{P}(G), \tag{34}$$

and

$$\|\psi(\boldsymbol{\omega})\|_2^2 = 2 \cdot \frac{(1-\epsilon^2)}{2} + \frac{\epsilon^2}{k} \cdot \sum_{i=2}^{p-1} \omega_i = 1.$$

Second, for all pairs of points $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}_\epsilon$,

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 = \delta_H(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j) \cdot \frac{\epsilon^2}{k} \leq 2 \cdot k \cdot \frac{\epsilon^2}{k} = 2 \cdot \epsilon^2.$$

The inequality follows from the fact that $\delta_H(\boldsymbol{\omega}, \mathbf{0}) = k+2$ $\omega_1 = 1$ and $\omega_p = 1$, $\forall \boldsymbol{\omega} \in \Omega_\xi$, and in turn

$$\delta_H(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j) \leq 2 \cdot k.$$

Similarly, for all pairs $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}_\epsilon$, $\mathbf{x}_i \neq \mathbf{x}_j$,

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2 = \delta_H(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j) \cdot \frac{\epsilon^2}{k} \geq \frac{1}{2} \cdot k \cdot \frac{\epsilon^2}{k} = \frac{\epsilon^2}{2},$$

where the inequality is due to (32). Finally, the lower bound on the cardinality of $\mathcal{X}_\epsilon$ follows immediately from (33) and the fact that $|\mathcal{X}_\epsilon| = |\Omega_\xi|$, which completes the proof. □

8. Details in proof of Lemma 1

We want to show that if

$$\epsilon^2 = \min\left\{1,\ \frac{C' \cdot (1+\beta)}{\beta^2} \cdot \frac{\log \frac{p-2}{k} + \frac{k}{4} \cdot \log d}{n}\right\},$$

for an appropriate choice of $C' > 0$, then the following two conditions (Eq. (13)) are satisfied:

$$n \cdot \frac{2\epsilon^2\beta^2}{(1+\beta)} \frac{1}{\log|\mathcal{X}_\epsilon|} \leq \frac{1}{4} \quad \text{and} \quad \log|\mathcal{X}_\epsilon| \geq 4\log 2.$$

For the second inequality, recall that by Lemma 2.2,

$$\log|\mathcal{X}_\epsilon| \ \geq\ \log\frac{p-2}{k} + \frac{1}{4} \cdot k\log d \ >\ 0. \tag{35}$$

Under the assumptions of Thm. 1 on the parameters k and d (note that $p - 2 \geq k \cdot d$ by the structure of G),

$$\log|\mathcal{X}_\epsilon| \geq \log\frac{p-2}{k} + \frac{k}{4} \cdot \log d \geq 4 \cdot H(3/4) \geq 4\log 2,$$

which is the desired result.

For the first inequality, we consider two cases:

- First, we consider the case where $\epsilon^2 = 1$, *i.e.*,

  $$\epsilon^2 = 1 \leq \frac{C' \cdot (1+\beta)}{\beta^2} \cdot \frac{\log \frac{p-2}{k} + \frac{k}{4} \cdot \log d}{n}.$$

  Equivalently,

  $$n \cdot \frac{2\epsilon^2\beta^2}{(1+\beta)} \leq 2 \cdot C' \cdot \left(\log\frac{p-2}{k} + \frac{k}{4} \cdot \log d\right). \tag{36}$$

- In the second case,

  $$\epsilon^2 = \frac{C' \cdot (1+\beta)}{\beta^2} \cdot \frac{\log \frac{p-2}{k} + \frac{k}{4} \cdot \log d}{n},$$

  which implies that

  $$n \cdot \frac{2\epsilon^2\beta^2}{(1+\beta)} = 2 \cdot C' \cdot \left(\log\frac{p-2}{k} + \frac{k}{4} \cdot \log d\right). \tag{37}$$

Combining (36) or (37), with (35), we obtain

$$n \cdot \frac{2\epsilon^2\beta^2}{(1+\beta)} \frac{1}{\log|\mathcal{X}_\epsilon|} \ \leq\ 2 \cdot C' \ \leq\ \frac{1}{4}$$

for $C' \leq 1/8$.

We conclude that for ϵ chosen as in (12), the conditions in (13) hold.

9. Other

Assumption 1. *There exist i.i.d. random vectors $\mathbf{z}_1, \dots, \mathbf{z}_n \in \mathbb{R}^p$, such that $\mathbb{E}\mathbf{z}_i = \mathbf{0}$ and $\mathbb{E}\mathbf{z}_i\mathbf{z}_i^\top = \mathbb{I}_p$,*

$$\mathbf{y} = \mu + \mathbf{\Sigma}^{1/2}\mathbf{z}_i \tag{38}$$

and

$$\sup_{\mathbf{x}\in\mathbb{S}_2^{p-1}} \|\mathbf{z}_i^\top \mathbf{x}\|_{\psi_2} \leq K, \tag{39}$$

where $\mu \in \mathbb{R}^p$ and $K > 0$ is a constant depending on the distribution of $\mathbf{z}_i$s.

10. NP-Hardness

Let GraphPathSPCA denote the decision version of the constrained quadratic maximization problem in (3):

> GraphPathSPCA
> **Input**: PSD matrix $\mathbf{A} \in \mathbb{S}^{n\times n}$, DAG G on n vertices (with auxiliary source and sink vertices S and T), threshold $\rho \in \mathbb{R}_+$.
> **Question**: Is there $\mathbf{x} \in \mathbb{R}^n$ with $\|\mathbf{x}\|_2 = 1$ and $\text{supp}(\mathbf{x}) \in \mathcal{P}(G)$[a], such that $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq \rho$?
>
> [a] $\mathcal{P}(G)$ denotes the collection of S-T paths in G. Since G is a DAG, an S-T path corresponds uniquely to a set of vertices in G, excluding S and T.

We show that GraphPathSPCA is NP-Complete via a reduction from the kCLIQUE, *i.e.*, the problem of determining whether a given arbitrary undirected graph G contains a clique of size k (k being part of the input). We perform the reduction in multiple steps. First we show that a seemingly special case of the kCLIQUE decision problem, referred to as kPkCLIQUE is also NP-Complete. In kPkCLIQUE, ones seeks to determine whether a k-partite undirected graph G contains a k-clique and the hardness is shown by a reduction from from the general kCLIQUE problem itself. We proceed with a reduction from kPkCLIQUE to MultiChoicePCA, a variant of PCA in which variables are subdivided into disjoint groups and the solution must contain at most one active variable from each group. Finally, we show that MultiChoicePCA is only a special case of GraphPathSPCA, which concludes the proof.

10.1. k-clique in k-partite graphs

> kCLIQUE
> **Input**: Undirected graph $G = (V, E)$, $k \in \mathbb{N}$.
> **Question**: Does G contain a k-clique?

The kCLIQUE is a well known NP-Complete problem. In the sequel, we consider an (at least seemingly) special case of the kCLIQUE problem.

Def. 1. *A k-partite graph $G = (V_1, \dots, V_k, E)$ is a graph whose vertices can be partitioned into k disjoint sets $V_1, \dots, V_k$, so that no two vertices within the same set are adjacent.*

In other words, each of the k vertex subsets $V_1, \dots, V_k$ in the k-partite graph G forms an independent set. The absence of edges within each set V_i, $i = 1, \dots, k$, implies that any clique in G can contain at most one vertex from each set, and the clique number $\omega(G)$ is upper bounded by k.

Let kPkCLIQUE be the problem of deciding whether a given undirected k-partite graph G *with known vertex partition* $V_1, \dots, V_k$, has a k-clique, *i.e.*, a clique comprising a vertex from each of the sets V_i, $i = 1, \dots, k$:

> kPkCLIQUE
> **Input**: Undirected k-partite graph $G = (V_1, \dots, V_k, E)$ along with the vertex partition $V_1, \dots, V_k$.
> **Question**: Does G contain a k-clique?

The kPkCLIQUE problem is NP-Complete by a reduction from kCLIQUE. kPkCLIQUE is in NP: given a set S of k vertices it can be verified in polynomial time whether these vertices form a k-clique. For the reduction, given an input $[G = (V, E),\ k]$ for kCLIQUE, consider the undirected k-partite graph $\widehat{G} = (\widehat{V}, \widehat{E})$ with vertex set

$$\widehat{V} \triangleq V \times [k] = \{(v, i) :\ v \in V, i \in [k]\},$$

partitioned into k sets

$$\widehat{V}_i \triangleq \left\{(v, i) \in \widehat{V} : v \in V\right\}, \quad i = 1, \dots, k.$$

An edge between (v, i) and (u, j) exists and only if

$$v \neq u \ \wedge\ i \neq j \ \wedge\ (v, u) \in E$$

which renders $\widehat{G}$ k-partite with partition $\{V_i\}_{i=1}^k$.

$\widehat{G}$ contains a k-clique if and only if G contains a k-clique. If G contains a k-clique among vertices $v_1, \dots, v_k \in V$, then $(v_1, 1), \dots, (v_k, k) \in \widehat{V}$ form a k-clique in $\widehat{G}$. Conversely, if $\widehat{G}$ contains a k-clique, the latter must contain exactly one vertex from each group $\widehat{V}_i$ and hence must be of the form $(v_1, 1), \dots, (v_k, k)$ for some $v_1, \dots, v_k \in V$ implying that the latter form a k-clique in G. Finally, note that $\widehat{G}$ is constructed in time polynomial in k and the size of G. We conclude that kPkCLIQUE is NP-Complete.

10.2. Multiple Choice SPCA

> MultiChoicePCA
> **Input**: PSD matrix $\mathbf{A} \in \mathbb{S}^{n\times n}$, a partition of the n variables into k disjoint sets $P_1, \dots, P_k$, and threshold $\rho \in \mathbb{R}_+$.
> **Question**: Is there $\mathbf{x} \in \mathbb{R}^n$ with $\|\mathbf{x}\|_2 = 1$ and $|\text{supp}(\mathbf{x}) \cap P_i| \leq 1\ \forall i \in [k]$, such that $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq \rho$?

MultiChoicePCA is a constrained version of the vanilla PCA problem, *i.e.*, the problem maximizing the Rayleigh quotient on an $n \times n$ PSD matrix. In MultiChoicePCA, the n variables are subdivided into k classes and the solution of the quadratic maximization can contain at most one nonzero variable from each class. We show that MultiChoicePCA is NP-Complete via a reduction from kPkCLIQUE.

Consider an undirected k-partite graph $G = (V, E)$ on n vertices, with known vertex partition $V_1, \ldots, V_k$; $V = \bigcup_{i=1}^{k} V_i$, and $V_i \cap V_j = \emptyset$, $\forall i, j \in [k]$, $i \neq j$. Assume an arbitrary labeling $1, \ldots, n$ of the vertices, and let $\mathbf{A}$ denote the adjacency matrix of G. For any $S \subset V$, let $\mathbf{A}_S$ denote the principal submatrix of $\mathbf{A}$ corresponding to S, *i.e.*, the $|S| \times |S|$ adjacency of the subgraph induced by S.

Lemma 10.5. *Let $\mathbf{A}$ be the adjacency matrix of a graph G on k vertices. If G is the complete graph (an k-clique), then $\lambda_1(\mathbf{A}) = k - 1$ with corresponding eigenvector $1/\sqrt{k} \cdot \mathbf{1}$. Otherwise, $\lambda_1(\mathbf{A}) < k - 1$.*

Proof. See Section 10.4 □

Consider the constrained quadratic maximization

$$\mathsf{OPT}_{\text{kSPARSE}} \triangleq \max_{\mathbf{x} \in \mathcal{X}_{\text{kSPARSE}}} \mathbf{x}^\top \mathbf{A} \mathbf{x}, \tag{40}$$

where

$$\mathcal{X}_{\text{kSPARSE}} \triangleq \left\{ \mathbf{x} : \|\mathbf{x}\|_2 = 1, \, \|\mathbf{x}\|_0 \leq k \right\}.$$

Let $\mathbf{x}_\star \in \mathcal{X}_{\text{kSPARSE}}$ denote the solution of (40) and $S_\star = \text{supp}(\mathbf{x}_\star)$. The objective value attained at $\mathbf{x}_\star$ is $\mathsf{OPT}_{\text{kSPARSE}} = \lambda_1(\mathbf{A}_{S_\star})$ By Lemma 10.5, $\lambda_1(\mathbf{A}_S) \leq k - 1$, $\forall S \subset V$, $|S| = k$, with equality achieved if and only if S forms a k-clique. We conclude that *G contains a k-clique if and only if $\mathsf{OPT}_{\text{kSPARSE}} = k - 1$.*

The above decision criterion based on (40) holds for arbitrary graphs. Here, G is k-partite. Any k-clique in G (if one exists) will contain exactly one vertex from each of the sets $V_1, \ldots, V_k$; if $S_\star$ forms a k-clique then $|S_\star \cap V_i| = 1$, $\forall i \in [k]$. Hence, we can explicitly enforce the constrain $|\text{supp}(\mathbf{x}) \cap V_i| \leq 1$, $\forall i \in [k]$ in (40) without affecting the decision criterion. Let

$$\text{OPT}_{\text{kPART}} \triangleq \max_{\mathbf{x} \in \mathcal{X}_{\text{kPART}}} \mathbf{x}^\top \mathbf{A} \mathbf{x}, \tag{41}$$

with

$$\mathcal{X}_{\text{kPART}} \triangleq \left\{ \mathbf{x} : \|\mathbf{x}\|_2 = 1, \, |\text{supp}(\mathbf{x}) \cap V_i| \leq 1 \, \forall i \in [k] \right\},$$

Then, *G has a k-clique if and only if $OPT_{\text{kPART}} = k - 1$.*

The quadratic maximization in (41) closely resembles the MultiChoicePCA problem, but does not satisfy the restriction that the input argument $\mathbf{A}$ must be a PSD matrix. In fact, $\mathbf{A}$ is the adjacency of a graph and will not be PSD. However, we can equivalently solve the quadratic maximization on $\mathbf{A}' \triangleq \mathbf{A} + |\lambda_n(\mathbf{A})| \cdot \mathbf{I}$, where $\lambda_n(\mathbf{A})$ is the smallest eigenvalue of $\mathbf{A}$. The new matrix $\mathbf{A}'$ is PSD and can be obtained from $\mathbf{A}$ in polynomial time. Further,

$$\mathbf{x}^\top \mathbf{A}' \mathbf{x} = \mathbf{x}^\top \mathbf{A} \mathbf{x} + |\lambda_n(\mathbf{A})|,$$

$\forall \mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_2 = 1$. In other words, substituting $\mathbf{A}$ with $\mathbf{A}'$ shifts the objective in (41) by a constant.

In summary, given an undirected k-partite graph G on n vertices with known vertex partition $V_1, \ldots, V_k$ and adjacency matrix $\mathbf{A}$ as input to kPkCLIQUE, we can decide on the existence of a k-clique in G by solving the MultiChoicePCA problem on the PSD matrix $\mathbf{A}' \triangleq \mathbf{A} + |\lambda_n(\mathbf{A})| \cdot \mathbf{I}$ and variable partition $V_1, \ldots, V_k$, and threshold $\rho = k - 1 + |\lambda_n(\mathbf{A})|$ Then, *G contains a k-clique if and only if the solution to MultiChoicePCA is equal to the threshold.*

10.3. From Multiple Choice to Graph-Path Sparse PCA

We conclude this section by noting that MultiChoicePCA is a special case of GraphPathSPCA, and show that the latter is also NP-Complete.

Consider a MultiChoicePCA input $[\mathbf{A}, \{P_1, \ldots, P_k\}, \rho]$. We construct a graph G on n vertices corresponding to the n variables of MultiChoicePCA (dimension of $\mathbf{A}$) partitioned into k disjoint sets $P_1, \ldots, P_k$. G contains a directed edge from each vertex of P_i to all vertices of P_{i+1}, $i = 1, \ldots, k - 1$. Finally, we conceptually augment G with two auxiliary vertices S (source) and T (terminal) and directed edges from S to all vertices in P_1 and directed edges from all vertices of P_k to T.

Any set Q of k variables containing exactly one variable from each set P_i, $i = 1, \ldots, k$, corresponds to an S-T path in G. Conversely, any S-T path in G corresponds to a set of variables with a unique representative from each set P_i, $i = 1, \ldots, k$. Therefore, the constraints in the maximization problems of the two instances are operationally identical: the two problems share a common set of feasible solutions and yield the same objective value for each such feasible vector. Hence, for any threshold ρ, MultiChoicePCA outputs *yes* if and only if GraphPathSPCA outputs *yes* for the corresponding input. We conclude that MultiChoicePCA can be polynomially reduced to GraphPathSPCA. GraphPathSPCA is clearly in NP, and hence it is NP-Complete.

10.4. Proof of Lemma 10.5

Consider an arbitrary labeling $1, \ldots, k$ of the k vertices in G. For any graph G, $\lambda_1(\mathbf{A}) \leq \max_{i \in [k]} d(i)$, where $d(i)$

denotes the degree of vertex i. To verify that, let $\mathbf{u} \neq \mathbf{0}$ be the eigenvector of $\mathbf{A}$ corresponding to the largest eigenvalue. By the Perron-Frobenius theorem $\mathbf{u} \geq \mathbf{0}$. Without loss of generality, let u_j be the largest entry of $\mathbf{u}$. Then,

$$\lambda_1(\mathbf{A}) = \frac{[\mathbf{A}\mathbf{u}]_j}{u_j} = \sum_{i \in \Gamma(j)} \frac{u_i}{u_j} \leq d(j) \leq k-1, \qquad (42)$$

where $\Gamma(i)$ denotes the neighborhood of vertex i.

If G is the complete graph on k vertices, the adjacency matrix is $\mathbf{A} = \mathbf{1}\mathbf{1}^\top - \mathbf{I}_k$. In that case, $k^{-1/2} \cdot \mathbf{1}$ is an eigenvector of $\mathbf{A}$ achieving the former upper bound. We conclude that if G is the complete graph, then $\lambda_1(\mathbf{A}) = k-1$ with corresponding eigenvector $\mathbf{u} = k^{-1/2} \cdot \mathbf{1}$.

It remains to show that if G is *not* the complete graph, then $\lambda_1(\mathbf{A}) < k-1$. Equivalently, we show that if $\lambda_1(\mathbf{A}) = k-1$, then G *is* the complete graph.

Without loss of generality, we can assume that G is a connected graph. If G is not connected, then $\mathbf{A}$ can be brought (with suitable row/column permutation) into a block diagonal form, where each block corresponds to a connected component. The eigenvalues of $\mathbf{A}$ are obtained by putting together the eigenvalues of the individual blocks. By (42), the largest eigenvalue of a block is upper bounded by $l-1$, where l is the dimension of the block, while the existence of multiple components implies that $l < k$.

Assume that $\lambda_1(\mathbf{A}) = k-1$ and let $\mathbf{u}$ be the corresponding eigenvector. As before, let j be the index of the largest entry in $\mathbf{u}$. By (42), we conclude that $d(j) = k-1$ and $u_j = u_i, \forall i \in \Gamma(j)$. But the fact that $d(j) = k-1$ implies that $\Gamma(j) = [k]$ and in turn that all entries of $\mathbf{u}$ are equal. Hence, repeating the argument of (42) on i (instead of j),

$$k-1 = \lambda_1(\mathbf{A}) = \frac{[\mathbf{A}\mathbf{u}]_i}{u_i} \leq d(i) \leq k-1 \quad \forall i \in [k].$$

We conclude that G is the complete graph.