Nonnegative Sparse PCA with Provable Guarantees

Megasthenis Asteris MEGAS@UTEXAS.EDU
Dimitris S. Papailiopoulos DIMITRIS@UTEXAS.EDU
Alexandros G. Dimakis DIMAKIS@AUSTIN.UTEXAS.EDU
Department of Electrical and Computer Engineering, The University of Texas at Austin, TX, USA

Abstract

We introduce a novel algorithm to compute nonnegative sparse principal components of positive semidefinite (PSD) matrices. Our algorithm comes with approximation guarantees contingent on the spectral profile of the input matrix $\mathbf{A}$: the sharper the eigenvalue decay, the better the quality of the approximation.

If the eigenvalues decay like any asymptotically vanishing function, we can approximate nonnegative sparse PCA within any accuracy ϵ in time polynomial in the matrix dimension n and desired sparsity k, but not in $^1/_\epsilon$. Further, we obtain a data dependent bound that is computed by executing an algorithm on a given data set. This bound is significantly tighter than *a-priori* bounds and can be used to show that for all tested datasets our algorithm is provably within $40\% - 90\%$ from the unknown optimum.

Our algorithm is combinatorial and explores a subspace defined by the leading eigenvectors of $\mathbf{A}$. We test our scheme on several data sets, showing that it matches or outperforms the previous state of the art.

1. Introduction

Given a data matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$ comprising m zero-mean vectors on n features, the first principal component (PC) is

$$\arg \max_{\|\mathbf{x}\|_2 = 1} \mathbf{x}^T \mathbf{A} \mathbf{x}, \tag{1}$$

where $\mathbf{A} = {}^1/_m \cdot \mathbf{S}\mathbf{S}^T$ is the $n \times n$ positive semidefinite (PSD) empirical covariance matrix. Subsequent PCs can be computed after $\mathbf{A}$ has been appropriately deflated to remove the first eigenvector. PCA is arguably the workhorse of high dimensional data analysis and achieves dimensionality reduction by computing the directions of maximum variance. Typically, all n features affect positively or negatively these directions resulting in dense PCs, which explain the largest possible data variance, but are often not interpretable.

It has been shown that enforcing nonnegativity on the computed principal components can aid interpretability. This is particularly true in applications where features interact only in an additive manner. For instance, in bioinformatics, chemical concentrations are nonnegative (Kim & Park, 2007), or the expression level of genes is typically attributed to positive or negative influences of those genes, but not both (Badea & Tilivea, 2005). Here, enforcing nonnegativity, in conjunction with sparsity on the computed components can assist the discovery of local patterns in the data. In computer vision, where features may coincide with non negatively valued image pixels, nonnegative sparse PCA pertains to the extraction of the most informative image parts (Lee & Seung, 1999). In other applications, nonnegative weights admit a meaningful probabilistic interpretation.

Sparsity emerges as an additional desirable trait of the computed components because it further helps interpretability (Zou et al., 2006; d'Aspremont et al., 2007b), even independently of nonnegativity. From a machine learning perspective, enforcing sparsity serves as an unsupervised feature selection method: the active coordinates in an optimal l_0-norm constrained PC should correspond to the most informative subset of features. Although nonnegativity inherently promotes sparsity, an explicit sparsity constraint enables precise control on the number of selected features.

Nonnegative Sparse PC. Nonnegativity and sparsity can be directly enforced on the principal component optimization by adding constraints to (1). The k-sparse nonnegative principal component of $\mathbf{A}$ is

$$\mathbf{x}_\star = \arg \max_{\mathbf{x} \in \mathbb{S}_k^n} \mathbf{x}^T \mathbf{A} \mathbf{x}, \tag{2}$$

Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32. Copyright 2014 by the author(s).

where $\mathbb{S}_k^n = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_2 = 1, \|\mathbf{x}\|_0 \leq k, \mathbf{x} \geq 0\}$, for a desired sparsity parameter $k \in [n]$.

The problem of computing the first eigenvector (1) is easily solvable, but with the additional sparsity and nonnegativity constraints problem (2) becomes computationally intractable. The cardinality constraint alone renders sparse PCA NP-hard (Moghaddam et al., 2006b). Even if the l_0-norm constraint is dropped, we show that problem (2) remains computationally intractable by reducing it to checking matrix copositivity, a well known co-NP complete decision problem (Murty & Kabadi, 1987; Parrilo, 2000). Therefore, each of the constraints $\mathbf{x} \geq \mathbf{0}$ and $\|\mathbf{x}\|_0 \leq k$ individually makes the problem intractable.

Our Contribution: We introduce a novel algorithm for approximating the nonnegative k-sparse principal component with provable approximation guarantees.

Given any PSD matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, sparsity parameter k, and accuracy parameter $d \in [n]$, our algorithm outputs a nonnegative, k-sparse, unit norm vector $\mathbf{x}_d$ that achieves at least ρ_d fraction of the maximum objective value in (2), *i.e.*,

$$\mathbf{x}_d^T \mathbf{A} \mathbf{x}_d \;\geq\; \rho_d \cdot \mathbf{x}_\star{}^T \mathbf{A} \mathbf{x}_\star, \tag{3}$$

where

$$\rho_d \;\geq\; \max\left\{\frac{k}{2n}, \frac{1}{1 + 2\frac{n}{k}\lambda_{d+1}/\lambda_1}\right\}. \tag{4}$$

Here, λ_i is the i^{th} largest eigenvalue of $\mathbf{A}$, and the accuracy parameter d specifies the rank of the approximation used and controls the running time. Specifically, our algorithm runs in time $O(n^d k^d + n^{d+1})$. As can be seen our result depends on the spectral profile of $\mathbf{A}$: the faster the eigenvalue decay, the tighter the approximation.

Near-Linear time approximation. Our algorithm has a running time $O(n^d k^d + n^{d+1})$, which in the linear sparsity regime can be as high as $O(n^{2d})$. This can be non-practical for large data sets, even if we set the rank parameter d to be two or three. We present a modification of our algorithm that can provably approximate the result of the first in near-linear time. Specifically, for any desired accuracy $\epsilon \in (0, 1]$ it computes a nonnegative, k-sparse, unit norm vector $\widehat{\mathbf{x}}_d$ such that

$$\widehat{\mathbf{x}}_d^T \mathbf{A} \widehat{\mathbf{x}}_d \;\geq\; (1-\epsilon) \cdot \rho_d \cdot \mathbf{x}_\star{}^T \mathbf{A} \mathbf{x}_\star, \tag{5}$$

where ρ_d is as described in (4). We show that the running time of our approximate algorithm is $O\left(\epsilon^{-d} \cdot n \log n\right)$, which is near-linear in n for any fixed accuracy parameters d and ϵ.

Our approximation theorem has several implications.

Exact solution for low-rank matrices. Observe that if the matrix $\mathbf{A}$ has rank d, our algorithm returns the optimal k-sparse PC for any target sparsity k. The same holds in the case of the rank-d update matrix $\mathbf{A} = \sigma \mathbf{I} + \mathbf{C}$, with $\text{rank}(\mathbf{C}) = d$ and arbitrary constant σ, since the algorithm can be equivalently applied on $\mathbf{C}$.

PTAS for any spectral decay. Consider the linear sparsity regime $k = c \cdot n$ and assume that the eigenvalues follow a decay law $\lambda_i \leq \lambda_1 \cdot f(i)$ for any decay function $f(i)$ which vanishes: $f(i) \to 0$ as $i \to \infty$. Special cases include power law decay $f(i) = 1/i^\alpha$ or even very slow decay functions like $f(i) = 1/\log\log i$. For all these cases, we can solve nonnegative sparse PCA for any desired accuracy ϵ in time polynomial in n and k, but not in $1/\epsilon$. Therefore, we obtain a polynomial-time approximation scheme (PTAS) for any spectral decay behavior.

Computable upper bounds. In addition to these theoretical guarantees, our method yields a data dependent upper bound on the maximum value of (2), that can be computed by running our algorithm. As it can be seen in Fig. 4-6, the obtained upper bound, combined with our achievable point, sandwiches the unknown optimum within a narrow region. Using this upper bound we are able to show that our solutions are within $40 - 90\%$ from the optimal in all the datasets that we examine. To the best of our knowledge, this framework of data dependent bounds has not been considered in the previous literature.

1.1. Related Work

There is a substantial volume of work on sparse PCA, spanning a rich variety of approaches: from early heuristics in (Jolliffe, 1995), to the LASSO based techniques in (Jolliffe et al., 2003), the elastic net l_1-regression in (Zou et al., 2006), a greedy branch-and-bound technique in (Moghaddam et al., 2006a), or semidefinite programming approaches (d'Aspremont et al., 2008; Zhang et al., 2012; d'Aspremont et al., 2007a). This line of work does not consider or enforce nonnegativity constraints.

When nonnegative components are desired, fundamentally different approaches have been used. Nonnegative matrix factorization (Lee & Seung, 1999) and its sparse variants (Hoyer, 2004; Kim & Park, 2007) fall within that scope: data is expressed as (sparse) nonnegative linear combinations of (sparse) nonnegative parts. These approaches are interested in finding a lower dimensionality representation of the data that reveals latent structure and minimizes a reconstruction error, but are not explicitly concerned with the statistical significance of individual output vectors.

Nonnegativity as an additional constraint on (sparse) PCA first appeared in (Zass & Shashua, 2007). The authors suggested a coordinate-descent scheme that jointly computes a set of nonnegative sparse principal components, maximizing the cumulative explained variance. An l_1-penalty promotes sparsity of computed components on average, but not on each component individually. A second convex penalty is incorporated to favor orthogonal components.

Similar convex optimization approaches for nonnegative PCA have been subsequently proposed in the literature. In (Allen & Maletić-Savatić, 2011) for instance, the authors suggest an alternating maximization scheme for the computation of the first nonnegative PC, allowing the incorporation of known structural dependencies.

A competitive algorithm for nonnegative sparse PCA was established in (Sigg & Buhmann, 2008), with the development of a framework stemming from Expectation-Maximization (EM) for a probabilistic generative model of PCA. The proposed algorithm, which enforces hard sparsity, or nonnegativity, or both constraints simultaneously, computes the first approximate PC in $O(n^2)$, *i.e.*, time quadratic in the number of features.

To the best of our knowledge, no prior works provide provable approximation guarantees for the nonnegative sparse PCA optimization problem. Further, no data dependent upper bounds have been present in the previous literature.

Differences from SPCA work. Our work is closely related to (Karystinos & Liavas, 2010; Asteris et al., 2011; Papailiopoulos et al., 2013) that introduced the ideas of solving low-rank quadratic combinatorial optimization problems on low-rank PSD matrices using hyperspectral transformations. Such transformations are called spannograms and follow a similar architecture. In this paper, we extend the spannogram framework to nonnegative sparse PCA. The most important technical issue compared to (Asteris et al., 2011; Papailiopoulos et al., 2013) is introducing nonnegativity constraints in spannogram algorithms.

To understand how this changes the problem, notice that in the original sparse PCA problem without nonnegativity constraints, if the support is known, the optimal principal component supported on that set can be easily found. However, under nonnegativity constraints, the problem is hard even if the optimal support is known. This is the fundamental technical problem that we address in this paper. We show that if the involved subspace is low-dimensional, it is possible to solve this problem.

2. Algorithm Overview

Given an $n \times n$ PSD matrix $\mathbf{A}$, the desired sparsity k, and an accuracy parameter $d \in [n]$, our algorithm computes a *nonnegative, k-sparse, unit norm* vector $\mathbf{x}_d$ approximating the nonnegative, k-sparse PC of $\mathbf{A}$. We begin with a high-level description of the main steps of the algorithm.

Step 1. *Compute $\mathbf{A}_d$, the rank-d approximation of* $\mathbf{A}$. We compute $\mathbf{A}_d$, the best rank-d approximation of $\mathbf{A}$, zeroing out the $n-d$ trailing eigenvalues of $\mathbf{A}$, that is,

$$\mathbf{A}_d = \sum_{i=1}^{d} \lambda_i \mathbf{u}_i \mathbf{u}_i^T,$$

where λ_i is the i^{th} largest eigenvalue of $\mathbf{A}$ and $\mathbf{u}_i$ the corresponding eigenvector.

Algorithm 1 Spannogram Nonnegative Sparse PCA

```
input  $\mathbf{A}$ ($n \times n$ PSD matrix), $k \in [n]$, $d \in [n]$.
$\mathbf{U}, \mathbf{\Lambda} \leftarrow \texttt{svd}(\mathbf{A}, d)$
$\mathbf{V} = \mathbf{U}\mathbf{\Lambda}^{1/2}$                                  $\{\mathbf{A}_d = \mathbf{V}\mathbf{V}^T\}$
$\mathcal{S}_d \leftarrow$ Spannogram($\mathbf{V}$, $k$)                          {Algo. 2}
$\mathcal{X}_d \leftarrow \{\}$                                      $\{|\mathcal{S}_d| \leq O(n^d)\}$
for all $\mathcal{I} \in \mathcal{S}_d$ do
    $\mathbf{c}^{(\mathcal{I})} \leftarrow \arg\max_{\substack{\|\mathbf{c}\|_2=1 \\ \mathbf{V}_\mathcal{I}\mathbf{c} \geq \mathbf{0}}} \|(\mathbf{V}_\mathcal{I}\mathbf{c})\|_2^2$        {Sec. 5}
    $\mathbf{x}_\mathcal{I}^{(\mathcal{I})} \leftarrow |\mathbf{V}_\mathcal{I}\mathbf{c}|/\|\mathbf{V}_\mathcal{I}\mathbf{c}\|$,   $\mathbf{x}_{\mathcal{I}^c}^{(\mathcal{I})} \leftarrow \mathbf{0}$
    $\mathcal{X}_d \leftarrow \mathcal{X}_d \cup \{\mathbf{x}^{(\mathcal{I})}\}$
end for                                                $\{|\mathcal{X}_d| \leq |\mathcal{S}_d|\}$
output  $\mathbf{x}_d \leftarrow \arg\max_{\mathbf{x} \in \mathcal{X}_d} \mathbf{x}^T \mathbf{A}_d \mathbf{x}$
```

Step 2. *Compute $\mathcal{S}_d$, a set of $O(n^d)$ candidate supports.* Enumerating the $\binom{n}{k}$ possible supports for k-sparse vectors in $\mathbb{R}^n$ is computationally intractable. Using our *Spannogram* technique described in Section 4, we efficiently determine a collection $\mathcal{S}_d$ of support sets, with cardinality $|\mathcal{S}_d| \leq 2^d\binom{n+1}{d}$, that provably contains the support of the nonnegative, k-sparse PC of $\mathbf{A}_d$.

Step 3. *Compute $\mathcal{X}_d$, a set of candidate solutions.* For each candidate support set $\mathcal{I} \in \mathcal{S}_d$, we compute a candidate solution $\mathbf{x}$ supported only in $\mathcal{I}$:

$$\arg \max_{\substack{\|\mathbf{x}\|_2=1, \mathbf{x} \geq \mathbf{0}, \\ \text{supp}(\mathbf{x}) \subseteq \mathcal{I}}} \mathbf{x}^T \mathbf{A}_d \mathbf{x}. \tag{6}$$

The constant rank of $\mathbf{A}_d$ is essential in solving (6): the constrained quadratic maximization is in general NP-hard, even for a given support.

Step 4. *Output the best candidate solution in $\mathcal{X}_d$, i.e.*, the candidate that maximizes the quadratic form.

If multiple components are desired, the procedure is repeated after an appropriate deflation has been applied on $\mathbf{A}_d$ (Mackey, 2008). The steps are formally presented in Algorithm 1. A detailed description is the subject of subsequent sections.

2.1. Approximation Guarantees

Instead of the nonnegative, k-sparse, principal component $\mathbf{x}_\star$ of $\mathbf{A}$, which attains the optimal value $\mathsf{OPT} = {\mathbf{x}_\star}^T\mathbf{A}\mathbf{x}_\star$, our algorithm outputs a nonnegative, k-sparse, unit norm vector $\mathbf{x}_d$. We measure the quality of $\mathbf{x}_d$ as a surrogate of $\mathbf{x}_\star$ by the approximation factor ${\mathbf{x}_d}^T\mathbf{A}\mathbf{x}_d/\mathsf{OPT}$. Clearly, the approximation factor takes values in $(0, 1]$, with higher values implying tighter approximation.

Theorem 1. *For any $n \times n$ PSD matrix $\mathbf{A}$, sparsity parameter k, and accuracy parameter $d \in [n]$, Alg. 1 outputs a nonnegative, k-sparse, unit norm vector $\mathbf{x}_d$ such that*

$$\mathbf{x}_d{}^T \mathbf{A} \mathbf{x}_d \;\geq\; \rho_d \cdot \mathbf{x}_\star{}^T \mathbf{A} \mathbf{x}_\star,$$

where

$$\rho_d \;\geq\; \max\left\{ \frac{k}{2n},\; \frac{1}{1 + 2\frac{n}{k}\lambda_{d+1}/\lambda_1} \right\},$$

in time $O(n^{d+1} + n^d k^d)$.

The approximation guarantee of Theorem 1 relies on establishing connections among the eigenvalues of $\mathbf{A}$, and the quadratic forms $\mathbf{x}_d{}^T \mathbf{A} \mathbf{x}_d$ and $\mathbf{x}_d{}^T \mathbf{A}_d \mathbf{x}_d$. The proof can be found in the supplemental material. The complexity of Algorithm 1 follows upon its detailed description.

3. Proposed Scheme

Our algorithm approximates the nonnegative, k-sparse PC of a PSD matrix $\mathbf{A}$ by computing the corresponding PC of $\mathbf{A}_d$, a rank-d surrogate of the input argument $\mathbf{A}$:

$$\mathbf{A}_d = \sum_{i=1}^{d} \mathbf{v}_i \mathbf{v}_i{}^T = \mathbf{V}\mathbf{V}^T, \tag{7}$$

where $\mathbf{v}_i = \sqrt{\lambda_i}\mathbf{u}_i$ is the scaled eigenvector corresponding to the i^{th} largest eigenvalue of $\mathbf{A}$, and $\mathbf{V} = [\mathbf{v}_1 \cdots \mathbf{v}_d] \in \mathbb{R}^{n \times d}$. In this section, we delve into the details of our algorithmic developments and describe how the low rank of $\mathbf{A}_d$ unlocks the computation of the desired PC.

3.1. Rank-1: A simple case

We begin with the rank-1 case because, besides its motivational simplicity, it is a fundamental component of the algorithmic developments for the rank-d case.

In the rank-1 case, $\mathbf{V}$ reduces to a single vector in $\mathbb{R}^n$ and $\mathbf{x}_1$, the nonnegative k-sparse PC of $\mathbf{A}_1$, is the solution to

$$\max_{\mathbf{x} \in \mathbb{S}_k^n} \mathbf{x}^T \mathbf{A}_1 \mathbf{x} = \max_{\mathbf{x} \in \mathbb{S}_k^n} \left(\mathbf{v}^T \mathbf{x}\right)^2. \tag{8}$$

That is, $\mathbf{x}_1$ is the nonnegative, k-sparse, unit length vector that maximizes $(\mathbf{v}^T\mathbf{x})^2$. Let $\mathcal{I} = \text{supp}(\mathbf{x}_1)$, $|\mathcal{I}| \leq k$, be the unknown support of $\mathbf{x}_1$. Then, $(\mathbf{v}^T\mathbf{x})^2 = \left(\sum_{i \in \mathcal{I}} v_i \cdot x_i\right)^2$. Since $\mathbf{x}_1 \geq \mathbf{0}$, it should not be hard to see that the active entries of $\mathbf{x}_1$ must correspond to nonnegative or nonpositive entries of $\mathbf{v}$, but not a combination of both. In other words, $\mathbf{v}_\mathcal{I}$, the entries of $\mathbf{v}$ indexed by $\mathcal{I}$, must satisfy $\mathbf{v}_\mathcal{I} \geq \mathbf{0}$ or $\mathbf{v}_\mathcal{I} \leq \mathbf{0}$. In either case, by the Cauchy-Schwarz inequality,

$$\left(\mathbf{v}^T \mathbf{x}\right)^2 = \left(\mathbf{v}_\mathcal{I}{}^T \mathbf{x}_\mathcal{I}\right)^2 \leq \|\mathbf{v}_\mathcal{I}\|_2^2 \|\mathbf{x}_\mathcal{I}\|_2^2, = \|\mathbf{v}_\mathcal{I}\|_2^2. \tag{9}$$

Equality in (9) can always be achieved by setting $\mathbf{x}_\mathcal{I} = \mathbf{v}_\mathcal{I}/\|\mathbf{v}_\mathcal{I}\|_2$ if $\mathbf{v}_\mathcal{I} \geq \mathbf{0}$, and $\mathbf{x}_\mathcal{I} = -\mathbf{v}_\mathcal{I}/\|\mathbf{v}_\mathcal{I}\|_2$ if $\mathbf{v}_\mathcal{I} \leq \mathbf{0}$. The support of the optimal solution $\mathbf{x}_1$ is the set $\mathcal{I}$ for which $\|\mathbf{v}_\mathcal{I}\|_2^2$ in (9) is maximized under the restriction that the entries of $\mathbf{v}_\mathcal{I}$ do not have mixed signs.

Def. 1. *Let $\mathcal{I}_k^+(\mathbf{v})$, $1 \leq k \leq n$ denote the set of indices of the (at most) k largest nonnegative entries in $\mathbf{v} \in \mathbb{R}^n$.*

Proposition 3.1. *Let $\mathbf{x}_1$ be the solution to problem (8). Then, supp $(\mathbf{x}_1) \in \mathcal{S}_1 = \left\{\mathcal{I}_k^+(\mathbf{v})\,,\; \mathcal{I}_k^+(-\mathbf{v})\right\}$.*

The collection $\mathcal{S}_1$ and the associated candidate vectors via (9) are constructed in $O(n)$. The solution $\mathbf{x}_1$ is the candidate that maximizes the quadratic.

3.2. Rank-d case

In the rank-d case, $\mathbf{x}_d$, the nonnegative, k-sparse PC of $\mathbf{A}_d$ is the solution to the following problem:

$$\max_{\mathbf{x} \in \mathbb{S}_k^n} \mathbf{x}^T \mathbf{A}_d \mathbf{x} = \max_{\mathbf{x} \in \mathbb{S}_k^n} \|\mathbf{V}^T \mathbf{x}\|_2^2. \tag{10}$$

Consider an auxiliary vector $\mathbf{c} \in \mathbb{R}^d$, with $\|\mathbf{c}\|_2 = 1$. From the Cauchy-Schwarz inequality,

$$\|\mathbf{V}^T\mathbf{x}\|_2^2 = \|\mathbf{c}\|_2^2 \|\mathbf{V}^T\mathbf{x}\|_2^2 \geq \left|\mathbf{c}^T \left(\mathbf{V}^T \mathbf{x}\right)\right|^2. \tag{11}$$

Equality in (11) is achieved if and only if $\mathbf{c}$ is colinear to $\mathbf{V}^T\mathbf{x}$. Since $\mathbf{c}$ spans the entire unit sphere, such a $\mathbf{c}$ exists for every $\mathbf{x}$, yielding an alternative description for the objective function in (10):

$$\|\mathbf{V}^T\mathbf{x}\|_2^2 = \max_{\mathbf{c} \in \mathbb{S}^d} \left|\left(\mathbf{V}\mathbf{c}\right)^T \mathbf{x}\right|^2, \tag{12}$$

where $\mathbb{S}^d = \left\{\mathbf{c} \in \mathbb{R}^d : \|\mathbf{c}\|_2 = 1\right\}$ is the d-dimensional unit sphere. The maximization in (10) becomes

$$\begin{aligned} \max_{\mathbf{x} \in \mathbb{S}_k^n} \|\mathbf{V}^T\mathbf{x}\|_2^2 &= \max_{\mathbf{x} \in \mathbb{S}_k^n} \max_{\mathbf{c} \in \mathbb{S}^d,} |\left(\mathbf{V}\mathbf{c}\right)^T \mathbf{x}|^2 \\ &= \max_{\mathbf{c} \in \mathbb{S}^d,\, \mathbf{x} \in \mathbb{S}_k^n} |\left(\mathbf{V}\mathbf{c}\right)^T \mathbf{x}|^2. \end{aligned} \tag{13}$$

The set of candidate supports. A first key observation is that for fixed $\mathbf{c}$, the product $(\mathbf{V}\mathbf{c})$ is a vector in $\mathbb{R}^n$. Maximizing $|\left(\mathbf{V}\mathbf{c}\right)^T \mathbf{x}|^2$ over all vectors $\mathbf{x} \in \mathbb{S}_k^n$ is a rank-1 instance of the optimization problem, as in (8). Let $(\mathbf{c}_d, \mathbf{x}_d)$ be the optimal solution of (10). By Proposition 3.1, the support of $\mathbf{x}_d$ coincides with either $\mathcal{I}_k^+(\mathbf{V}\mathbf{c}_d)$ or $\mathcal{I}_k^+(-\mathbf{V}\mathbf{c}_d)$. Hence, we can safely claim that supp$(\mathbf{x}_d)$ appears in

$$\mathcal{S}_d = \bigcup_{\mathbf{c} \in \mathbb{S}^d} \left\{\mathcal{I}_k^+(\mathbf{V}\mathbf{c})\right\}. \tag{14}$$

Naively, one might think that $\mathcal{S}_d$ can contain as many as $\binom{n}{k}$ distinct support sets. In Section 4, we show that $|\mathcal{S}_d| \leq 2^d\binom{n+1}{d}$ and present our Spannogram technique (Alg. 2) for efficiently constructing $\mathcal{S}_d$ in $O(n^{d+1})$. Each support in $\mathcal{S}_d$ corresponds to a candidate principal component.

Solving for a given support. We seek a pair $(\mathbf{x}, \mathbf{c})$ that maximizes (13) under the additional constraint that $\mathbf{x}$ is supported only on a given set $\mathcal{I}$. By the Cauchy-Schwarz inequality, the objective in (13) satisfies

$$|\left(\mathbf{V}\mathbf{c}\right)^T \mathbf{x}|^2 = |\left(\mathbf{V}_{\mathcal{I}}\mathbf{c}\right)^T \mathbf{x}_{\mathcal{I}}|^2 \leq \|\left(\mathbf{V}_{\mathcal{I}}\mathbf{c}\right)\|_2^2, \tag{15}$$

where $\mathbf{V}_{\mathcal{I}}$ is the matrix formed by the rows of $\mathbf{V}$ indexed by $\mathcal{I}$. Equality in (15) is achieved if and only if $\mathbf{x}_{\mathcal{I}}$ is colinear to $\mathbf{V}_{\mathcal{I}}\mathbf{c}$. However, it is not achievable for arbitrary $\mathbf{c}$, as $\mathbf{x}_{\mathcal{I}}$ must be nonnegative. From Proposition 3.1, we infer that $\mathbf{x}$ being supported in $\mathcal{I}$ implies that all entries of $\mathbf{V}_{\mathcal{I}}\mathbf{c}$ have the same sign. Further, whenever the last condition holds, a nonnegative $\mathbf{x}_{\mathcal{I}}$ colinear to $\mathbf{V}_{\mathcal{I}}\mathbf{c}$ exists and equality in (15) can be achieved. Under the additional constraint that $\text{supp}(\mathbf{x}) = \mathcal{I} \in \mathcal{S}_d$, the maximization in (13) becomes

$$\max_{\mathbf{c}\in\mathbb{S}^d} \max_{\substack{\mathbf{x}\in\mathbb{S}_k^n \\ \text{supp}(\mathbf{x})\subseteq\mathcal{I}}} |\left(\mathbf{V}\mathbf{c}\right)^T \mathbf{x}|^2 = \max_{\substack{\mathbf{c}\in\mathbb{S}^d \\ \mathbf{V}_{\mathcal{I}}\mathbf{c}\geq\mathbf{0}}} \|\left(\mathbf{V}_{\mathcal{I}}\mathbf{c}\right)\|_2^2. \tag{16}$$

The constraint $\mathbf{V}_{\mathcal{I}}\mathbf{c} \geq \mathbf{0}$ in (16), is equivalent to requiring that all entries in $\mathbf{V}_{\mathcal{I}}\mathbf{c}$ have the same sign, since $\mathbf{c}$ and $-\mathbf{c}$ achieve the same objective value.

The optimization problem in (16) is NP-hard. In fact, it encompasses the original nonnegative PCA problem as a special case. Here, however, the constant dimension $d = \Theta(1)$ of the unknown variable $\mathbf{c}$ permits otherwise intractable operations. In Section 5, we outline an $O(k^d)$ algorithm for solving this constrained quadratic maximization.

The algorithm. The previous discussion suggests a two-step algorithm for solving the rank-d optimization problem in (10). First, run the Spannogram algorithm to construct $\mathcal{S}_d$, the collection of $O(n^d)$ candidate supports for $\mathbf{x}_d$, in $O(n^{d+1})$. For each $\mathcal{I} \in \mathcal{S}_d$, solve (16) in $O(k^d)$ to obtain a candidate solution $\mathbf{x}^{(\mathcal{I})}$ supported on $\mathcal{I}$. Output the candidate solution that maximizes the quadratic $\mathbf{x}^T\mathbf{A}_d\mathbf{x}$. Efficiently combining the previous steps yields an $O(n^{d+1} + n^d k^d)$ procedure for approximating the nonnegative sparse PC, outlined in Alg. 1.

4. The Nonnegative Spannogram

In this section, we describe how to construct $\mathcal{S}_d$, the collection of candidate supports, defined in (14) as

$$\mathcal{S}_d = \bigcup_{\mathbf{c}\in\mathbb{S}^d} \left\{\mathcal{I}_k^+(\mathbf{V}\mathbf{c})\right\},$$

for a given $\mathbf{V} \in \mathbb{R}^{n\times d}$. $\mathcal{S}_d$ comprises all support sets induced by vectors in the range of $\mathbf{V}$. The *Spannogram* of $\mathbf{V}$ is a *visualization* of its range, and a valuable tool in efficiently collecting those supports.

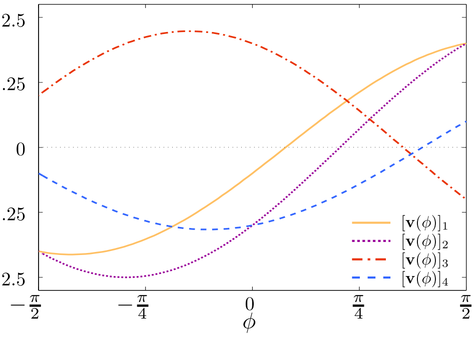

Figure 1. Spannogam of an arbitrary rank-2 matrix $\mathbf{V} \in \mathbb{R}^{4\times 2}$. At a point ϕ, the values of the curves correspond to the entries of a vector $\mathbf{v}(\phi)$ in the range of $\mathbf{V}$ and vice versa.

4.1. Constructing $\mathcal{S}_2$

We describe the $d = 2$ case, the simplest nontrivial case, to facilitate a gentle exposure to the Spannogram technique. The core ideas generalize to arbitrary d and a detailed description is provided in the supplemental material.

Spherical variables. Up to scaling, all vectors $\mathbf{v}$ in the range of $\mathbf{V} \in \mathbb{R}^{n\times 2}$, $\mathcal{R}(\mathbf{V})$, can be written as $\mathbf{v} = \mathbf{V}\mathbf{c}$ for some $\mathbf{c} \in \mathbb{R}^2 : \|\mathbf{c}\| = 1$. We introduce a variable $\phi \in \Phi = (-\pi/2, \pi/2]$, and set $\mathbf{c}$ to be the following function of ϕ:

$$\mathbf{c}(\phi) = \left[\sin(\phi)\ \cos(\phi)\right]^T.$$

The range of $\mathbf{V}$, $\mathcal{R}(\mathbf{V}) = \{\pm\mathbf{v}(\phi) = \pm\mathbf{V}\mathbf{c}(\phi), \phi \in \Phi\}$, is also a function of ϕ, and in turn $\mathcal{S}_2$ can be expressed as

$$\mathcal{S}_2 = \bigcup_{\phi\in\Phi} \left\{\mathcal{I}_k^+(\mathbf{v}(\phi)),\ \mathcal{I}_k^+(-\mathbf{v}(\phi))\right\}.$$

Spannogram. The i^{th} entry of $\mathbf{v}(\phi)$ is a continuous function of ϕ generated by the i^{th} row of $\mathbf{V}$: $[\mathbf{v}(\phi)]_i = \mathbf{V}_{i,1}\sin(\phi) + \mathbf{V}_{i,2}\cos(\phi)$. Fig. 1 depicts the functions corresponding to the rows of an arbitrary matrix $\mathbf{V} \in \mathbb{R}^{4\times 2}$. We call this a *spannogram*, because at each ϕ, the values of the curves coincide with the entries of a vector in the range of $\mathbf{V}$. A key observation is that the sorting of the curves at some ϕ is locally invariant for most points in Φ. In fact, due to the continuity of the curves, as we move along the ϕ-axis, the set $\mathcal{I}_k^+(\mathbf{v}(\phi))$ can only change at points where a curve intersects with *(i)* another curve, or *(ii)* the zero axis; a change in either the sign of a curve or the relative order of two curves is necessary, although not sufficient, for $\mathcal{I}_k^+(\mathbf{v}(\phi))$ to change.

Appending a zero $(n+1)^{\text{th}}$ row to $\mathbf{V}$, the two aforementioned conditions can be merged into one: $\mathcal{I}_k^+(\mathbf{v}(\phi))$ can change only at the points where two of the $n+1$ curves intersect. Finding the unique intersection point of two curves $[\mathbf{v}(\phi)]_i$ and $[\mathbf{v}(\phi)]_j$ for all pairs $\{i, j\}$ is the key to discovering all possible candidate support sets. There are exactly $\binom{n+1}{2}$ such points partitioning Φ into $\binom{n+1}{2}+1$ intervals within which the set of largest k nonnegative entries of $\mathbf{v}(\phi)$ and $-\mathbf{v}(\phi)$ are invariant.

Constructing $\mathcal{S}_2$. The point ϕ_{ij} where the i^{th} and j^{th} curves intersect, corresponds to a vector $\mathbf{v}(\phi_{ij}) \in \mathcal{R}(\mathbf{V})$ whose i^{th} and j^{th} entries are equal. To find it, it suffices to compute a $\mathbf{c} \neq \mathbf{0}$ such that $(\mathbf{e}_i - \mathbf{e}_j)^T\mathbf{V}\mathbf{c} = 0$, *i.e.*, a unit norm vector $\mathbf{c}_{ij}$ in the one-dimensional nullspace of $(\mathbf{e}_i - \mathbf{e}_j)^T\mathbf{V}$. Then, $\mathbf{v}(\phi_{ij}) = \mathbf{V}\mathbf{c}_{ij}$.

We compute the candidate support $\mathcal{I}_k^+(\mathbf{v}(\phi_{ij}))$ at the intersection. Assuming for simplicity that only the i^{th} and j^{th} curves intersect at ϕ_{ij}, the sorting of all curves is unchanged in a small neighborhood of ϕ_{ij}, except the i^{th} and j^{th} curves whose order changes over ϕ_{ij}. If both the i^{th} and j^{th} entries of $\mathbf{v}(\phi_{ij})$ or none of them is included in the k largest nonnegative entries, then the set $\mathcal{I}_k^+(\mathbf{v}(\phi))$ in the two intervals incident to ϕ_{ij} is identical. Otherwise, the i^{th} and j^{th} curve occupy the k^{th} and $(k+1)^{\text{th}}$ order at ϕ_{ij}, and the change in their relative order implies that one leaves and one joins the set of k largest nonnegative curves at ϕ_{ij}. The support sets associated with the two adjacent intervals differ only in one element (one contains index i and the other contains index j instead), while the remaining $k-1$ common indices correspond to the $k-1$ largest curves at the intersection point ϕ_{ij}. We include both in $\mathcal{S}_2$ and repeat the above procedure for $\mathcal{I}_k^+(-\mathbf{v}(\phi_{ij}))$.

Each pairwise intersection is computed in $O(1)$ and the at most 4 associated candidate supports in $O(n)$. In total, the collection $\mathcal{S}_2$ comprises $|\mathcal{S}_2| \leq 4\binom{n+1}{2} = O(n^2)$ candidate supports and can be constructed in $O(n^3)$.

The generalized Spannogram algorithm for constructing $\mathcal{S}_d$ runs in $O(n^{d+1})$ and is formally presented in Alg. 2. A detailed description is provided in the supplemental material.

5. Quadratic maximization over unit vectors in the intersection of halfspaces

Each support set $\mathcal{I}$ in $\mathcal{S}_d$ yields a candidate nonnegative, k-sparse PC, which can be obtained by solving (16), a quadratic maximization over the intersection of halfspaces and the unit sphere:

$$\mathbf{c}_\star = \arg\max_{\substack{\mathbf{c}\in\mathbb{S}^d \\ \mathbf{R}\mathbf{c}\geq 0}} \mathbf{c}^T\mathbf{Q}\mathbf{c}, \qquad (P_d)$$

where $\mathbf{Q} = \mathbf{V}_{\mathcal{I}}^T\mathbf{V}_{\mathcal{I}}$ is a $d \times d$ matrix and $\mathbf{R}$ is a $k \times d$ matrix. Problem (P_d) is NP-hard: for $\mathbf{Q}$ PSD and $\mathbf{R} = \mathbf{I}_{d\times d}$, it reduces to the original problem in (2). Here, however, we are interested in the case where the dimension d is a constant. We outline an $O(k^d)$ algorithm, *i.e.*, polynomial in the number of linear constraints, for solving (P_d). A detailed proof is available in the supplemental material.

Algorithm 2 Spannogram algorithm for constructing $\mathcal{S}_d$

```
input  $\mathbf{V} \in \mathbb{R}^{n\times d}$, $k \in [n]$.
$\mathcal{S}_d \leftarrow \{\}$                                   {Set of candidate supports in $\mathcal{R}(\mathbf{V})$}
$\widehat{\mathbf{V}} \leftarrow \left[\mathbf{V}^T\ \mathbf{0}_d\right]^T$                      {Append zero row; $\widehat{\mathbf{V}} \in \mathbb{R}^{n+1\times d}$}
for all $\binom{n+1}{d}$ sets $\{i_1,\dots,i_d\} \subseteq [n+1]$ do
    $\mathbf{c} \leftarrow \text{Nullspace}\left(\begin{bmatrix} \mathbf{e}_{i_1}^T - \mathbf{e}_{i_2}^T \\ \vdots \\ \mathbf{e}_{i_1}^T - \mathbf{e}_{i_d}^T \end{bmatrix} \widehat{\mathbf{V}} \in \mathbb{R}^{d-1\times d}\right)$
    for $\alpha \in \{+1,-1\}$ do
        $\mathcal{I} \leftarrow \mathcal{I}_k^+(\alpha\mathbf{V}\mathbf{c})$                    {Entries $\geq$ than the $k^{\text{th}}$ one}
        if $|\mathcal{I}| \leq k$ then
            $\mathcal{S}_d \leftarrow \mathcal{S}_d \cup \{\mathcal{I}\}$                          {No ambiguity}
        else
            $\mathcal{A} \leftarrow \{i_1,\dots,i_d\}\backslash\{n+1\}$            {Ambiguous set}
            $\mathcal{T} \leftarrow \mathcal{I}\backslash\{\{i_1,\dots,i_d\} \cup \{n+1\}\}$
            $r \leftarrow k - |\mathcal{T}|$
            $\widehat{d} \leftarrow |\mathcal{A}|$
            for all $\binom{\widehat{d}}{r}$ $r$-subsets $\mathcal{A}^{(r)} \subseteq \mathcal{A}\}$ do
                $\widehat{\mathcal{I}} \leftarrow \mathcal{T} \cup \mathcal{A}^{(r)}$
                $\mathcal{S}_d \leftarrow \mathcal{S}_d \cup \{\widehat{\mathcal{I}}\}$             {$\leq 2^d$ new candidates}
            end for
        end if
    end for
end for
output  $\mathcal{S}_d$
```

The objective of (P_d) is maximized by $\mathbf{u}_1 \in \mathbb{R}^d$, the leading eigenvector of $\mathbf{Q}$. If $\mathbf{u}_1$ or $-\mathbf{u}_1$ is feasible, *i.e.*, if it satisfies all linear constraints, then $\mathbf{c}_\star = \pm\mathbf{u}_1$. It can be shown that if none of $\pm\mathbf{u}_1$ is feasible, at least one of the k linear constraints is active at the optimal solution $\mathbf{c}_\star$, that is, there exists $1 \leq i \leq k$ such that $\mathbf{R}_{i,:}\mathbf{c}_\star = 0$.

Fig. 2 depicts an example for $d = 2$. The leading eigenvector of $\mathbf{Q}$ lies outside the feasible region, an arc of the unit-circle in the intersection of k halfspaces. The optimal solution coincides with one of the two endpoints of the feasible region, where a linear inequality is active, motivating a simple algorithm for solving (P_2): (*i*) for each linear inequality determine a unit length point where the inequality becomes active, and (*ii*) output the point that is feasible and maximizes the objective.

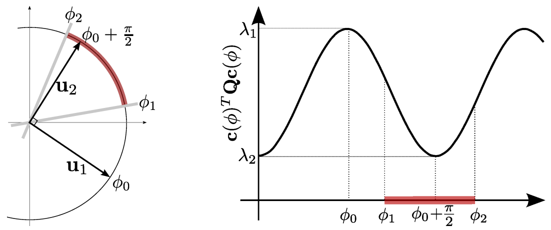

Figure 2. An instance of (P_2): the leading eigenvector $\mathbf{u}_1$ of $\mathbf{Q} \in \mathbb{S}^2$ lies outside the feasible region (highlighted arc). The maximum of the constrained quadratic optimization problem is attained at ϕ_1, an endpoint of the feasible interval.

Back to the general (P_d) problem, if a linear inequality $\mathbf{R}_{i,:}\mathbf{c} \geq 0$ for some $i \in [k]$ is enforced with equality, the modified problem can be written as a quadratic maximization in the form of (P_d), with dimension reduced to $d-1$ and $k-1$ linear constraints. This observation suggests a recursive algorithm for solving (P_d): If $\pm\mathbf{u}_1$ is feasible, it is also the optimal solution. Otherwise, for $i = 1, \ldots, k$, set the i^{th} inequality constraint active, solve recursively, and collect candidate solutions. Finally, output the candidate that maximizes the objective. The $O(k^d)$ recursive algorithm is formally presented in the supplemental material.

6. Near-Linear Time Nonnegative SPCA

Alg. 1 approximates the nonnegative, k-sparse PC of a PSD matrix $\mathbf{A}$ by solving the nonnegative sparse PCA problem exactly on $\mathbf{A}_d$, the best rank-d approximation of $\mathbf{A}$. Albeit polynomial in n, the running time of Alg. 1 can be impractical even for moderate values of n.

Instead of pursuing the exact solution to the low-rank nonnegative sparse PCA problem $\max_{\mathbf{x}\in\mathbb{S}_k^n} \mathbf{x}^T\mathbf{A}_d\mathbf{x}$, we can compute an approximate solution in near-linear time, with performance arbitrarily close to optimal. The suggested procedure is outlined in Algorithm 3, and a detailed discussion is provided in the supplemental material. Alg. 3 relies on randomly sampling points from the range of $\mathbf{A}_d$ and efficiently solving rank-1 instances of the nonnegative sparse PCA problem as described in Section 3.1.

Theorem 2. *For any $n \times n$ PSD matrix $\mathbf{A}$, sparsity parameter k, and accuracy parameters $d \in [n]$ and $\epsilon \in (0, 1]$, Alg. 3 outputs a nonnegative, k-sparse, unit norm vector $\widehat{\mathbf{x}}_d$ such that*

$$\widehat{\mathbf{x}}_d^T\mathbf{A}\widehat{\mathbf{x}}_d \;\geq\; (1-\epsilon)\cdot\rho_d\cdot{\mathbf{x}_\star}^T\mathbf{A}\mathbf{x}_\star,$$

with probability at least $1 - 1/n$, in time $O\left(\epsilon^{-d}\cdot n\log n\right)$ plus the time to compute the d leading eigenvectors of $\mathbf{A}$.

7. Experimental Evaluation

We empirically evaluate the performance of our algorithm on various datasets and compare it to the EM algorithm[1] for sparse and nonnegative PCA of (Sigg & Buhmann, 2008) which is known to outperform previous algorithms.

CBCL Face Dataset. The CBCL face image dataset (Sung, 1996), with 2429 gray scale images of size 19 × 19 pixels, has been used in the performance evaluation of both the NSPCA (Zass & Shashua, 2007) and EM (Sigg & Buhmann, 2008) algorithms.

```
Algorithm 3 Approximate Spannogram NSPCA ($\epsilon$-net)
input $\mathbf{A}$ ($n \times n$ PSD matrix), $k, d \in [n]$, $\epsilon \in (0,1]$
  $[\mathbf{U}, \mathbf{\Lambda}] = \mathtt{svd}(\mathbf{A}, d)$
  $\mathbf{V} = \mathbf{U}\mathbf{\Lambda}^{1/2}$                    $\{\mathbf{A}_d = \mathbf{V}\mathbf{V}^T\}$
  $\mathcal{X}_d = \emptyset$
  for $i = 1 : O(\epsilon^{-d}\cdot\log n)$ do
      $\mathbf{c} = \mathtt{randn}(d, 1)$
      $\mathbf{a} = \mathbf{V}\mathbf{c}/\|\mathbf{c}\|_2$
      $\mathbf{x} = \mathtt{rank1solver}(\mathbf{a})$          {Section 3.1}
      $\mathcal{X}_d = \mathcal{X}_d \cup \{\mathbf{x}\}$
  end for
output $\widehat{\mathbf{x}}_d = \arg\max_{\mathbf{x}\in\mathcal{X}_d} \|\mathbf{V}^T\mathbf{x}\|_2^2$
```

Fig. 3 depicts samples from the dataset, as well as six orthogonal, nonnegative, k-sparse components ($k = 40$) successively computed by *(i)* Alg. 3 ($d = 3, \epsilon = 0.1$) and *(ii)* the EM algorithm. Features active in one component are removed from the dataset prior to computing subsequent PCs to ensure orthogonality. Fig. 3 reveals the ability of nonnegative sparse PCA to extract significant parts.

In Fig. 4, we plot the variance explained by the computed approximate nonnegative, k-sparse PC (normalized by the leading eigenvalue) versus the sparsity parameter k. Alg. 3 for $d = 3$ and $\epsilon = 0.1$, and the EM algorithm exhibit nearly identical performance. For this dataset, we also compute the leading component using the NSPCA algorithm of (Zass & Shashua, 2007). Note that NSPCA does not allow for a precise control of the sparsity of its output; an appropriate sparsity penalty β was determined via binary search for each target sparsity k. We plot the explained variance only for those values of k for which a k-sparse component was successfully extracted. Finally, note that both the EM and NSPCA algorithms are randomly initialized. All depicted values are the best results over multiple random restarts.

Our theory allows us to obtain provable approximation guarantees: based on Theorem 2 and the output of Alg. 3, we compute a data dependent upper bound on the maximum variance, which provably lies in the shaded area. For instance, for $k = 180$, the extracted component explains at least 58% of the variance explained by the true nonnegative, k-sparse PC. The quality of the bound depends on the accuracy parameters d and ϵ, and the eigenvalue decay of the empirical covariance matrix of the data. There exist

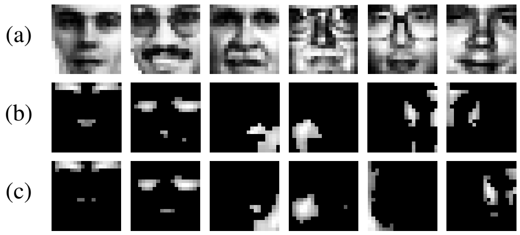

Figure 3. We plot (a) six samples from the dataset, and the six leading orthogonal, nonnegative, k-sparse PCs for $k = 40$ extracted by (b) Alg. 3 ($d = 3, \epsilon = 0.1$), and (c) the EM algorithm.

[1] Matlab implementation available by the author.

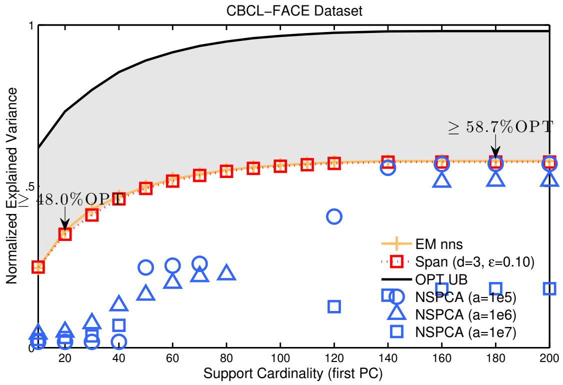

Figure 4. CBCL dataset (Sung, 1996). We plot the normalized variance explained by the approximate nonnegative, k-sparse PC versus the sparsity k. Our theory yields a provable data dependent approximation guarantee: the true unknown optimum provably lies in the shaded area.

datasets on which our algorithm provably achieves 70% or even 90% of the optimal.

Leukemia Dataset. The Leukemia dataset (Armstrong et al., 2001) contains 72 samples, each consisting of expression values for 12582 probe sets. The dataset was used in the evaluation of (Sigg & Buhmann, 2008). In Fig. 5, we plot the normalized variance explained by the computed nonnegative, k-sparse PC versus the sparsity parameter k. For low values of k, Alg. 3 outperforms the EM algorithm in terms of explained variance. For larger values, the two algorithms exhibit similar performance.

The approximation guarantees accompanying our algorithm allow us to upper bound the optimal performance. For k as small as 50, which roughly amounts to 0.4% of the features, the extracted component captures at least 44.6% of the variance corresponding to the true nonnegative k-sparse PC. The obtained upper bound is a significant improvement compared to the trivial bound given by λ_1.

Low Resolution Spectrometer Dataset. The Low Resolution Spectrometer (LRS) dataset, available in (Bache & Lichman, 2013), originates from the Infra-Red Astronomy Satellite Project. It contains 531 high quality spectra (samples) measured in 93 bands. Fig. 6 depicts the normalized variance explained by the computed nonnegative, k-sparse PC versus the sparsity parameter k. The empirical covariance matrix of this dataset exhibits sharper decay in the spectrum than the previous examples, yielding tighter approximation guarantees according to our theory. For instance, for $k = 20$, the extracted nonnegative component captures at least 86% of the maximum variance. For values closer to $k = 90$, where the computed PC is nonnegative but no longer sparse, this value climbs to nearly 93%.

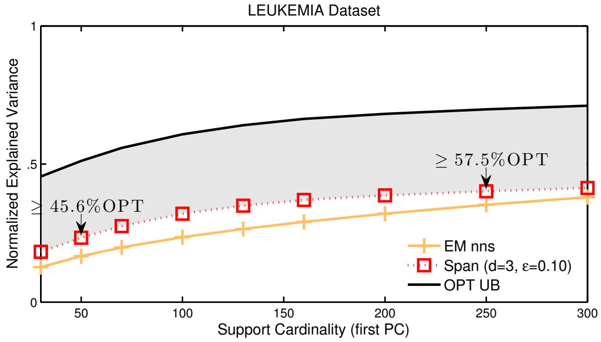

Figure 5. Leukemia dataset (Armstrong et al., 2001). We plot the normalized variance explained by the output of Alg. 3 ($d = 3, \epsilon = 0.1$) versus the sparsity k, and compare with the EM algorithm of (Sigg & Buhmann, 2008). By our approximation guarantees, the maximum variance provably lies in the shaded area.

8. Conclusions

We introduced a novel algorithm for nonnegative sparse PCA, expanding the spannogram theory to nonnegative quadratic optimization. We observe that the performance of our algorithm often matches and sometimes outperforms the previous state of the art (Sigg & Buhmann, 2008). Even though the theoretical running time of Alg. 3 scales better than EM, in practice we observed similar speed, both in the order of a few seconds. Our approach has the benefit of provable approximation, giving both theoretical a-priori guarantees and data dependent bounds that can be used to estimate the variance explained by nonnegative sparse PCs, as shown in our experiments.

9. Acknowledgements

The authors would like to acknowledge support from NSF grants CCF-1344364, CCF-1344179, DARPA XDATA and research gifts by Google and Docomo.

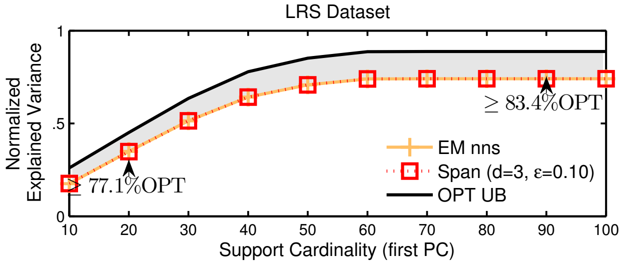

Figure 6. LRS dataset (Bache & Lichman, 2013). We plot the normalized explained variance versus the sparsity k. Alg. 3 ($d = 3, \epsilon = 0.1$) and the EM algorithm exhibit similar performance. The optimum value of the objective in (2) provably lies in the shaded area, which in this case is particularly tight.

References

Allen, Genevera I. and Maletić-Savatić, Mirjana. Sparse nonnegative generalized pca with applications to metabolomics. *Bioinformatics*, 2011.

Armstrong, Scott A, Staunton, Jane E, Silverman, Lewis B, Pieters, Rob, den Boer, Monique L, Minden, Mark D, Sallan, Stephen E, Lander, Eric S, Golub, Todd R, and Korsmeyer, Stanley J. Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nature genetics*, 30(1):41–47, 2001.

Asteris, M., Papailiopoulos, D.S., and Karystinos, G.N. Sparse principal component of a rank-deficient matrix. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pp. 673–677, 2011.

Bache, K. and Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

Badea, Liviu and Tilivea, Doina. Sparse factorizations of gene expression guided by binding data. In *Pacific Symposium on Biocomputing*, 2005.

d'Aspremont, A., El Ghaoui, L., Jordan, M.I., and Lanckriet, G.R.G. A direct formulation for sparse pca using semidefinite programming. *SIAM review*, 49(3):434–448, 2007a.

d'Aspremont, Alexandre, Bach, Francis R., and Ghaoui, Laurent El. Full regularization path for sparse principal component analysis. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, pp. 177–184, New York, NY, USA, 2007b. ACM.

d'Aspremont, Alexandre, Bach, Francis, and Ghaoui, Laurent El. Optimal solutions for sparse principal component analysis. *J. Mach. Learn. Res.*, 9:1269–1294, Jun 2008.

Hoyer, Patrik O. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5: 1457–1469, 2004.

Jolliffe, I.T. Rotation of principal components: choice of normalization constraints. *Journal of Applied Statistics*, 22(1):29–35, 1995.

Jolliffe, I.T., Trendafilov, N.T., and Uddin, M. A modified principal component technique based on the lasso. *Journal of Computational and Graphical Statistics*, 12(3):531–547, 2003.

Karystinos, G.N. and Liavas, A.P. Efficient computation of the binary vector that maximizes a rank-deficient quadratic form. *Information Theory, IEEE Transactions on*, 56(7):3581–3593, 2010.

Kim, Hyunsoo and Park, Haesun. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23(12): 1495–1502, 2007.

Lee, Daniel D and Seung, H Sebastian. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788–791, 1999.

Mackey, Lester. Deflation methods for sparse pca. In *Advances in Neural Information Processing Systems 21*, NIPS '08, pp. 1–8, Vancouver, Canada, Dec 2008.

Moghaddam, B., Weiss, Y., and Avidan, S. Spectral bounds for sparse pca: Exact and greedy algorithms. *Advances in neural information processing systems*, 18:915, 2006a.

Moghaddam, Baback, Weiss, Yair, and Avidan, Shai. Generalized spectral bounds for sparse lda. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pp. 641–648, 2006b.

Murty, Katta G. and Kabadi, Santosh N. Some np-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 39(2):117–129, 1987.

Papailiopoulos, D. S., Dimakis, A. G., and Korokythakis, S. Sparse pca through low-rank approximations. In *Proceedings of the 30th International Conference on Machine Learning*, ICML '13, pp. 767–774. ACM, 2013.

Parrilo, Pablo A. *Structured Semidenite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology Pasadena, California, 2000.

Sigg, Christian D. and Buhmann, Joachim M. Expectation-maximization for sparse and non-negative pca. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pp. 960–967, New York, NY, USA, 2008. ACM.

Sung, Kah-Kay. *Learning and example selection for object and pattern recognition*. PhD thesis, PhD thesis, MIT, Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Cambridge, MA, 1996.

Zass, Ron and Shashua, Amnon. Nonnegative sparse pca. In *Advances in Neural Information Processing Systems 19*, pp. 1561–1568, Cambridge, MA, 2007. MIT Press.

Zhang, Y., d'Aspremont, A., and Ghaoui, L.E. Sparse pca: Convex relaxations, algorithms and applications. *Handbook on Semidefinite, Conic and Polynomial Optimization*, pp. 915–940, 2012.

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