Sparse PCA via Bipartite Matchings

Megasthenis Asteris, Dimitris Papailiopoulos, Anastasios Kyrillidis, Alex Dimakis

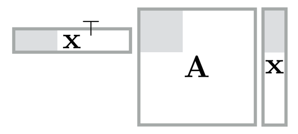
The University of Texas at Austin

[Sparse PCA]

Given a covariance matrix $\mathbf{A}$, find direction of maximum variance, as a linear combination of only a few variables:

Empirical Covariance

$$\mathbf{x}_\star = \arg\max_{\mathbf{x}\in\mathcal{X}} \left(\mathbf{x}^\top\mathbf{A}\mathbf{x}\right)$$

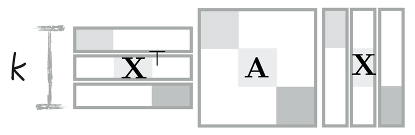


$$\mathcal{X} = \{\mathbf{x}\in\mathbb{R}^d : \|\mathbf{x}\|_2 = 1, \|\mathbf{x}\|_0 = s\}$$

Sparse vector (NP-hard)

[Multiple Sparse Components]

Find multiple sparse components with disjoint support sets:



(MultiSPCA) $\quad \mathbf{X}_\star = \arg\max_{\mathbf{X}\in\mathcal{X}_k} \mathrm{Tr}\left(\mathbf{X}^\top\mathbf{A}\mathbf{X}\right)$

Sparse columns

$$\mathcal{X}_k = \left\{ \begin{array}{l} \mathbf{X}\in\mathbb{R}^{d\times k} : \|\mathbf{X}^j\|_2 = 1, \|\mathbf{X}^j\|_0 = s, \ \forall j \\ \mathrm{supp}(\mathbf{X}^i)\cap\mathrm{supp}(\mathbf{X}^j) = \emptyset, \ \forall i,j \end{array} \right\}$$

Disjoint support sets

Example: NY Times text corpus
- Find 8 components, each 10-sparse.
- Sparse disjoint components interpreted as distinct topics.

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8
1:	percent	zzz_united_states	zzz_bush	company	team	cup	school	zzz_al_gore
2:	million	zzz_u_s	official	companies	game	minutes	student	zzz_george_bush
3:	money	zzz_american	government	market	season	add	children	campaign
4:	high	attack	president	stock	player	tablespoon	women	election
5:	program	military	group	business	play	oil	show	plan
6:	number	palestinian	leader	billion	point	teaspoon	book	tax
7:	need	war	country	analyst	run	water	family	public
8:	part	administration	political	firm	right	pepper	look	zzz_washington
9:	problem	zzz_white_house	american	sales	home	large	hour	member
10:	com	games	law	cost	won	food	small	nation

[One approach: Deflation]

Compute components one-by-one.
- Compute one sparse PC.
- Remove used variables from the dataset.
- Repeat.

Simple but, suboptimal.

Problem:
Given a 4 x 4 PSD matrix $\mathbf{A}$, find two 2-sparse components $\mathbf{x}_1, \mathbf{x}_2$ with disjoint supports, that maximize $\mathbf{x}_1^\top\mathbf{A}\mathbf{x}_1 + \mathbf{x}_2^\top\mathbf{A}\mathbf{x}_2$.

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \epsilon \\ 0 & \delta & 0 & 0 \\ 0 & 0 & \delta & 0 \\ \epsilon & 0 & 0 & 1 \end{bmatrix}$$

Solution I: Deflation

$$\begin{bmatrix} 1 & 0 & 0 & \epsilon \\ 0 & \delta & 0 & 0 \\ 0 & 0 & \delta & 0 \\ \epsilon & 0 & 0 & 1 \end{bmatrix} \quad \lambda_{\max}\left(\begin{bmatrix} 1 & \epsilon \\ \epsilon & 1 \end{bmatrix}\right) + \lambda_{\max}\left(\begin{bmatrix} \delta & 0 \\ 0 & \delta \end{bmatrix}\right) = 1 + \epsilon + \delta \ll 2$$

(suboptimal)

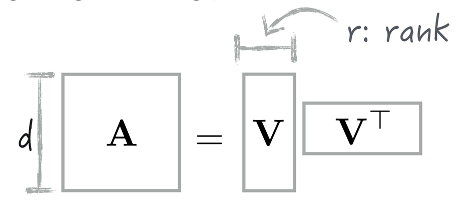
Solution II: Joint Optimization

$$\begin{bmatrix} 1 & 0 & 0 & \epsilon \\ 0 & \delta & 0 & 0 \\ 0 & 0 & \delta & 0 \\ \epsilon & 0 & 0 & 1 \end{bmatrix} \quad \lambda_{\max}\left(\begin{bmatrix} 1 & 0 \\ 0 & \delta \end{bmatrix}\right) + \lambda_{\max}\left(\begin{bmatrix} \delta & 0 \\ 0 & 1 \end{bmatrix}\right) = 1 + 1 = 2$$

[Our Algorithm]

Think of the d x d matrix $\mathbf{A}$ as having rank r. For now r < d.

Matrix $\mathbf{A}$ is PSD and can be decomposed into $\mathbf{A} = \mathbf{V}\mathbf{V}^\top$.



Observation I

$$\mathbf{x}^\top\mathbf{A}\mathbf{x} = \|\mathbf{V}^\top\mathbf{x}\|_2^2 \geq \langle\mathbf{V}^\top\mathbf{x}, \mathbf{c}\rangle^2 \qquad \forall \mathbf{c}\in\mathbb{R}^r : \|\mathbf{c}\|_2 = 1$$

In turn, a variational characterization is the following:

$$\mathbf{x}^\top\mathbf{A}\mathbf{x} = \max_{\mathbf{c}\in\mathbb{S}_2^{r-1}} \langle\mathbf{x}, \mathbf{V}\mathbf{c}\rangle^2$$

For multiple components…

$$\max_{\mathbf{X}\in\mathcal{X}_k} \mathrm{Tr}\left(\mathbf{X}^\top\mathbf{A}\mathbf{X}\right) = \max_{\mathbf{X}\in\mathcal{X}_k} \max_{\mathbf{C}:\mathbf{C}^j\in\mathbb{S}_2^{r-1}\forall j} \sum_{j=1}^{k} \langle\mathbf{X}^j, \mathbf{V}\mathbf{C}^j\rangle^2.$$

SPCA as a "double" maximization

Observation II
Fix the value of the $r\times k$ variable $\mathbf{C}$. Let $\mathbf{W}\leftarrow\mathbf{V}\mathbf{C}$.

$$\widehat{\mathbf{X}} = \arg\max_{\mathbf{X}\in\mathcal{X}_k} \sum_{j=1}^{k} \langle\mathbf{X}^j, \mathbf{W}^j\rangle^2$$

Can be solved. How? (Later)

→ SPCA reduces to determining the optimal $\mathbf{C}$.
→ Low dimensional variable: sample to find the best.

r x k variable. (smaller than X)

Algorithm

Input: $d\times d$ rank-r PSD $\mathbf{A}$
- Initialize empty collection $\mathcal{S}$
- Compute $\mathbf{V}\leftarrow\mathrm{Chol}(\mathbf{A})$ $\quad (d\times r)$
- For $i = 1 : O\left((4/\epsilon)^{r\cdot k}\right)$
  - Sample $\mathbf{C}$ (r x k variable. Each column is unit-norm)
  - Compute $\mathbf{W}\leftarrow\mathbf{V}\mathbf{C}$
  - Solve
  $$\widehat{\mathbf{X}} = \arg\max_{\mathbf{X}\in\mathcal{X}_k} \sum_{j=1}^{k} \langle\mathbf{X}^j, \mathbf{W}^j\rangle^2$$
  - Add $\widehat{\mathbf{X}}$ to the collection $\mathcal{S}$.

Output: Best solution $\overline{\mathbf{X}}$ in collection $\mathcal{S}$.

Theorem I: Algo Guarantees (Low rank)

Input: $i)$ $d\times d$ rank-r PSD matrix $\mathbf{A}$, $ii)$ k: desired # of components, $iii)$ s # nnz entries/component, $iv)$ accuracy $\epsilon\in(0,1)$.

Output: $\overline{\mathbf{X}}\in\mathcal{X}_k$ such that

$$\mathrm{Tr}\left(\overline{\mathbf{X}}^\top\mathbf{A}\overline{\mathbf{X}}\right) \geq (1-\epsilon)\cdot\mathsf{OPT},$$

in time $T_{\mathsf{SVD}}(r) + O\left(\left(\frac{4}{\epsilon}\right)^{r\cdot k}\cdot d\cdot(s\cdot k)^2\right)$.

[Subroutine]

$$\widehat{\mathbf{X}} = \arg\max_{\mathbf{X}\in\mathcal{X}_k} \sum_{j=1}^{k} \langle\mathbf{X}^j, \mathbf{W}^j\rangle^2$$

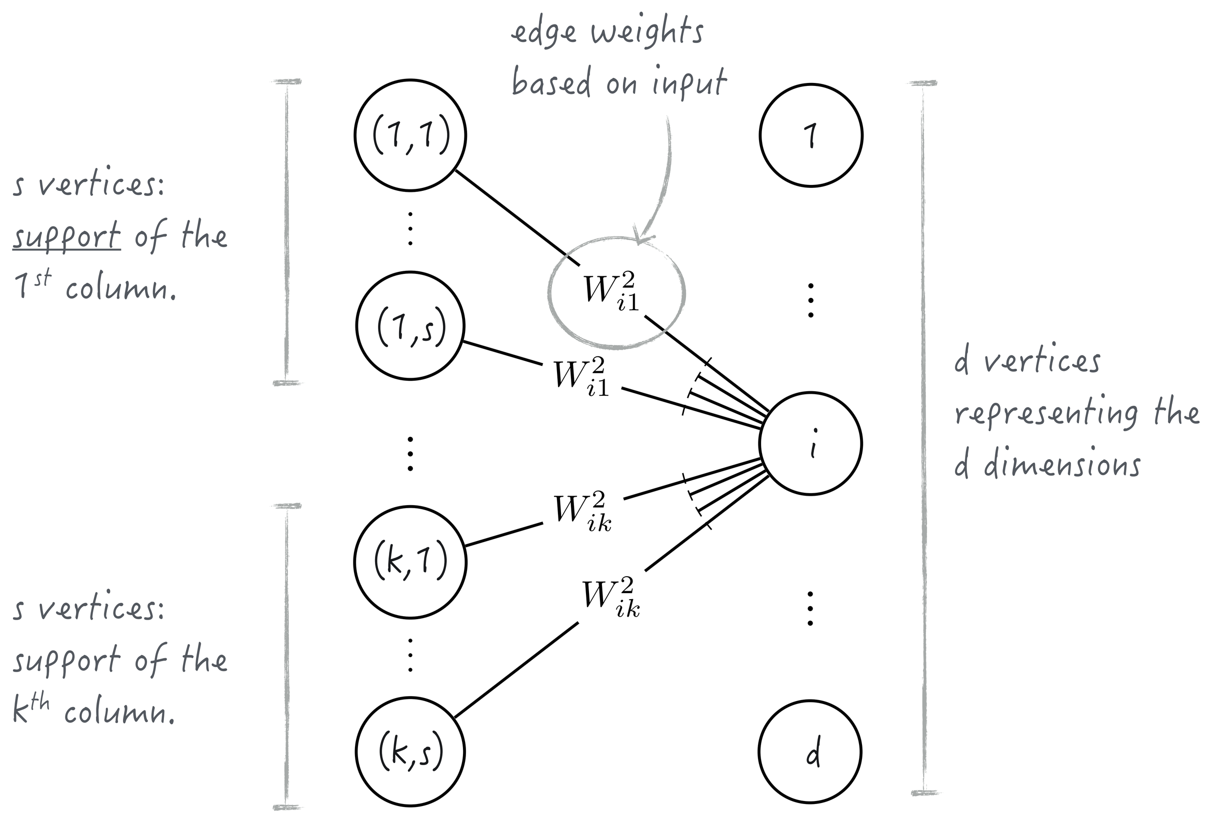
$\mathcal{I}_1,\ldots,\mathcal{I}_k$: disjoint support sets of the k components (columns of $\widehat{\mathbf{X}}$).

Observation I
If we knew the support sets $\mathcal{I}_1,\ldots,\mathcal{I}_k$, we could determine the optimal value based on Cauchy-Schwarz:

$$(\ast) \qquad \sum_{j=1}^{k} \left\langle\widehat{\mathbf{X}}^j, \mathbf{W}^j\right\rangle^2 = \sum_{j=1}^{k}\sum_{i\in\mathcal{I}_j} W_{ij}^2.$$

Unknown supports. Find them.

Consider the complete bipartite graph G on $k\cdot s + d$ vertices:



Maximum Weight Matching on G:
- Each vertex on the left is mapped to a vertex on the right.
  → s indices are assigned to each "support set".
- Each right vertex is used at most once.
  → Support sets are disjoint.
- Maximum weight = maximum objective in ($\ast$).

Algorithm

Input: $d\times k$ matrix $\mathbf{W}$
s : # nnz entries / column of $\widehat{\mathbf{X}}$)
1. Construct bipartite graph G as above.
2. Compute maximum weight matching to determine the supports $\mathcal{I}_1,\ldots,\mathcal{I}_k$
3. Compute each column of $\widehat{\mathbf{X}}$ for the given support based on Cauchy-Schwarz.

[Summary]

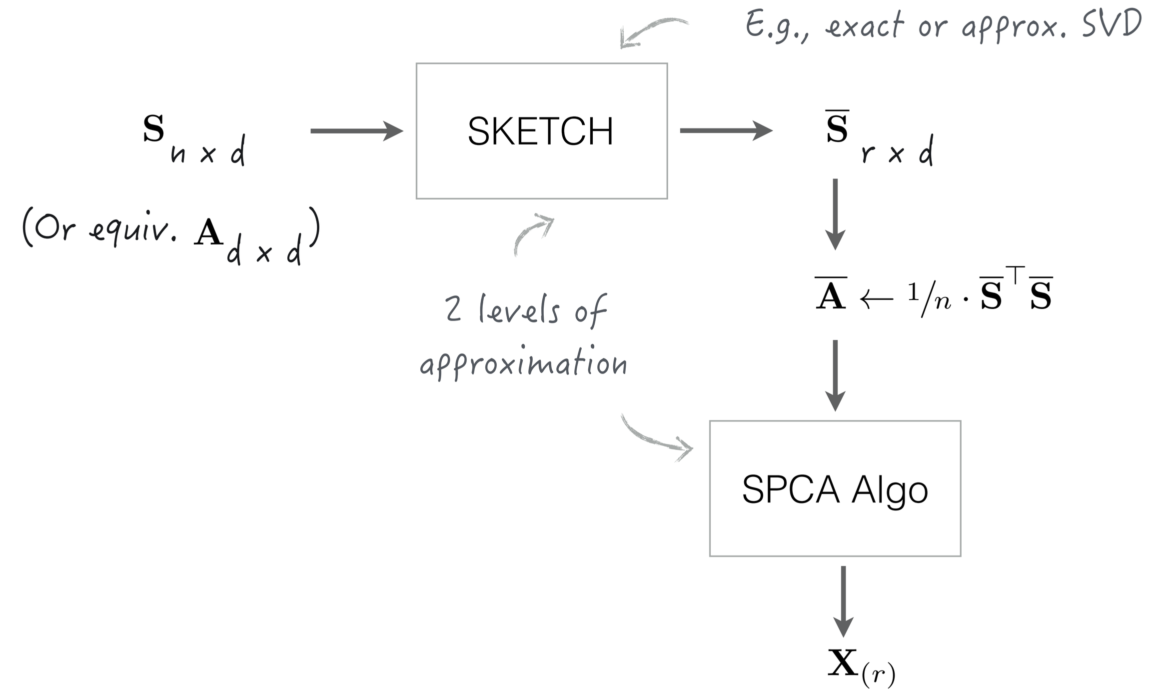
- First algorithm for multi-component SPCA with disjoint supports; Operates by recasting MultiSPCA into multiple instances of the **bipartite maximum weight matching** problem.
- Provable approximation guarantees.
- Complexity:
  - Low-order polynomial in the ambient dimension d, but
  - Exponential in the intrinsic dimension r.

Still much better than naive brute force.

Separates ambient and intrinsic dimension.

[SPCA on a Low Dim Sketch]

In reality, data is not low rank.
However, maybe **close to low rank.**
→ Spectrum of $\mathbf{A}$ may be sharply decaying
→ $\mathbf{A}$ is well approximated by a low rank matrix.



Theorem II: Algo Guarantees (Full rank)

Input: $i)$ $n\times d$ input data matrix $\mathbf{S}$ (or covariance $\mathbf{A} = 1/n\cdot\mathbf{S}^\top\mathbf{S}$) $ii)$ k: # of components, $iii)$ s # nnz entries/component, $iv)$ accuracy $\epsilon\in(0,1)$, $v)$ r: rank of approximation,

Output: $\mathbf{X}_{(r)}\in\mathcal{X}_k$ such that

$$\mathrm{Tr}\left(\mathbf{X}_{(r)}^\top\mathbf{A}\mathbf{X}_{(r)}\right) \geq (1-\epsilon)\cdot\mathsf{OPT} - 2\cdot k\cdot\|\mathbf{A}-\overline{\mathbf{A}}\|_2,$$

in time $T_{\mathsf{SKETCH}}(r) + T_{\mathsf{SVD}}(r) + O\left(\left(\frac{4}{\epsilon}\right)^{r\cdot k}\cdot d\cdot(s\cdot k)^2\right)$.

Extra time: for computing the sketch

Extra error: depends on the quality of the sketch.

[In Practice]

- Taking too long?
- Run our algorithm and stop it any time.
  → Ignore the theoretical guarantees
  → Still finds solutions with higher explained variance, compared to deflation based methods.

Example: Leukemia Dataset
- # samples n = 72, dimension d=12582 (probe sets)
- Compare to deflation using TPower, EM-SPCA and SpanSPCA for the single component SPCA problem.