A Randomized Rounding Algorithm for Sparse PCA

Eugenia-Maria Kontopoulou

in collaboration with

K. Fountoulakis, A. Kundu & P. Drineas

Department of Computer Science
Purdue University

PUNLAG Seminars

Purdue, April 2017

Principal Component Analysis (PCA)

Definition

Given a centered matrix $X \in \mathbb{R}^{m \times n}$ and the matrix $A = X^\top X$, we seek to find the vector w_{opt} that solves:

$$\begin{aligned} \underset{w \in \mathbb{R}^n}{\text{maximize}} \quad & w^\top A\, w \\ \text{subject to} \quad & \|w\|_2 = 1 \end{aligned} \tag{1}$$

Principal Component Analysis (PCA)

Definition

Given a centered matrix $X \in \mathbb{R}^{m \times n}$ and the matrix $A = X^\top X$, we seek to find the vector w_{opt} that solves:

$$\begin{aligned} \underset{w \in \mathbb{R}^n}{\text{maximize}} \quad & w^\top A\, w \\ \text{subject to} \quad & \|w\|_2 = 1 \end{aligned} \tag{1}$$

The objective function of Problem (1) is the Rayleigh Quotient, R, and for a Symmetric Positive Semidefinite matrix like A the maximum value of R is the dominant eigenvalue while w_{opt} is the corresponding eigenvector.

PCA Computation

- Singular Value Decomposition
- Eigenvalue Decomposition
- Krylov Methods (Lanczos etc)

Why not satisfied?

PCA Computation

- Singular Value Decomposition
- Eigenvalue Decomposition
- Krylov Methods (Lanczos etc)

But what happens in the case of Big Data?

Memory Issues

- entire matrix in RAM
- sparsity is not preserved

Data Interpretation Issues

- difficult direct interpretation

Why not satisfied?

PCA Computation

- Singular Value Decomposition
- Eigenvalue Decomposition
- Krylov Methods (Lanczos etc)

But what happens in the case of Big Data?

Memory Issues

- entire matrix in RAM
- sparsity is not preserved

Data Interpretation Issues

- difficult direct interpretation

Solution: Add a sparsity constraint in Problem 1!!!

Definition

Given a centered data matrix $X \in \mathbb{R}^{m \times n}$, the matrix $A = X^\top X$ and a parameter k, we seek to find the vector w_{opt} that solves:

$$\begin{aligned} \underset{w \in \mathbb{R}^n}{\text{maximize}} \quad & w^\top A\, w \\ \text{subject to} \quad & \|w\|_0 \leq k, \\ & \|w\|_2 = 1. \end{aligned} \tag{2}$$

- ✓ k enforces the sparsity of w_{opt}, (at most k non-zero entries).
- ✓ NP-hard if k grows with n.
- ✓ Non-convex constraints.
- ✓ **Common approaches:** thresholding the top singular vector, convex relaxations of the constraints, semi-definite programming, . . .

Sparse PCA

Fountoulakis, Kontopoulou, Kundu, Drineas, ACM TKDD (2017)

Definition

Given a centered data matrix $X \in \mathbb{R}^{m \times n}$, the matrix $A = X^\top X$ and a parameter k, we seek to find the vector w_{opt} that solves:

$$\begin{aligned} \underset{w \in \mathbb{R}^n}{\text{maximize}} \quad & w^\top A\, w \\ \text{subject to} \quad & \|w\|_1 \leq \sqrt{k}, \\ & \|w\|_2 \leq 1. \end{aligned} \tag{3}$$

- ✓ (convex) l_1 relaxation of the sparsity constraint.
- ✓ convex relaxation of the 2-norm constraint.

Two-step algorithm:

1. Compute a stationary point $\tilde{w}_{opt}$.
2. Invoke a randomized rounding strategy to compute $\hat{w}_{opt}$.

How we find the stationary point:

1. Compute the gradient and make a gradient step.
2. Project onto the l_1 ball with radius $\sqrt{k}$.
3. Repeat until a relative error threshold is reached.

Randomized rounding strategy:

Given $\tilde{w}_{opt}$, define each element of $\hat{w}_{opt}$ as follows (*opt* subscript is dropped):

$$\hat{w}_i = \begin{cases} \frac{1}{p_i}\tilde{w}_i & \text{with } p_i = \min\left\{\frac{s|\tilde{w}_i|}{||\tilde{w}||_1}, 1\right\} \\ 0, & \text{otherwise} \end{cases}$$

In (1) we prove the following Theorem

Theorem

Let w_{opt} be the optimal solution of the Sparse PCA problem (2) satisfying $\|w_{opt}\|_2 = 1$ and $\|w_{opt}\|_0 \leq k$. Let $\hat{w}_{opt}$ be the vector returned when the rounding sparsification strategy is applied on the optimal solution $\tilde{w}_{opt}$ of the optimization problem (3), with $s = 200k/\epsilon^2$, where $\epsilon \in (0, 1]$ is an accuracy parameter. Then, $\hat{w}_{opt}$ has the following properties:

1. $\mathbb{E}\|\hat{w}_{opt}\|_0 \leq s$.
2. *With probability at least 3/4,*
$$\|\hat{w}_{opt}\|_2 \leq 1 + 0.15\epsilon.$$
3. *With probability at least 3/4,*
$$\hat{w}_{opt}^{\top} A \hat{w}_{opt} \geq w_{opt}^{\top} A w_{opt} - \epsilon.$$

Proofs

Datasets

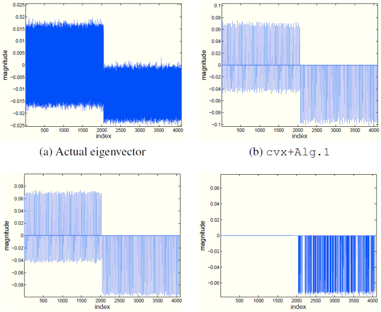
- **Synthetic:** $m = 2^7$, $n = 2^{12}$
- **Classic-2:** $m = 2,858$ documents , $n = 12,427$ terms
  1. CISI collection (1,460 information retrieval abstracts)
  2. CRANFIELD collection (1,398 aeronautical systems abstracts)

Evaluation

- $\|w\|_0/n \quad \text{vs} \quad f(w) = w^\top Aw/\|A\|_2$
- Pattern Captured
- Sparsity Captured
- Variance Captured

We test our algorithm (Naive & SVD-based) with other SPCA software like `MaxComp` (Naive & SVD-based) and `Spasm`.

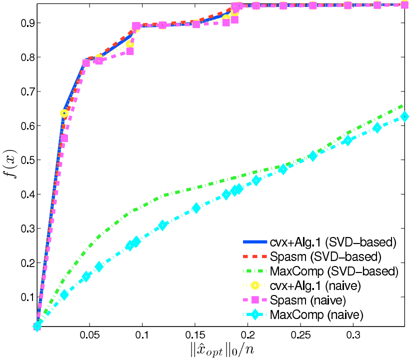
Pattern capture



(a) Actual eigenvector (b) `cvx+Alg.1`

(c) `Spasm` (d) `MaxComp`

Sparsity ratio vs Eigenvalue capture



`cvx` refers to the solution of the optimization problem and `Alg. 1` to the randomized rounding technique.

Real Data Application

Table 1: Variance and sparsity captured by the principal components. PCA results in dense principal components, while Spasm and MaxComp share the same sparsity with rspca.

	k	pca	cvx	rspca	MaxComp	Spasm
Top Principal Comp.	100	0.4351	0.3077 (99%)	0.2942 (99%)	0.1955	0.2768
Top two Principal Comp.		0.6802	0.4897 (99%)	0.4680 (99%)	0.3391	0.4227
Top Principal Comp.	500	0.4351	0.3880 (95%)	0.3728 (98%)	0.3353	0.3601
Top two Principal Comp.		0.6802	0.6073 (95%)	0.5864 (98%)	0.5399	0.5701
Top Principal Comp.	1000	0.4351	0.4136 (90%)	0.4005 (95%)	0.3825	0.3912
Top two Principal Comp.		0.6802	0.6486 (90%)	0.6294 (95%)	0.6074	0.6163
Top Principal Comp.	1500	0.4351	0.4242 (84%)	0.4120 (93%)	0.4013	0.4039
Top two Principal Comp.		0.6802	0.6649 (82%)	0.6470 (93%)	0.6342	0.6361
Top Principal Comp.	2000	0.4351	0.4295 (75%)	0.4190 (91%)	0.4133	0.4131
Top two Principal Comp.		0.6802	0.6730 (70%)	0.6572 (91%)	0.6503	0.6491
Top Principal Comp.	4000	0.4351	0.4350 (6%)	0.4278 (81%)	0.4284	0.4271
Top two Principal Comp.		0.6802	0.6801 (3%)	0.6700 (81%)	0.6710	0.6690
Top Principal Comp.	8000	0.4351	0.4351 (0%)	0.4324 (68%)	0.4326	0.4316
Top two Principal Comp.		0.6802	0.6802 (0%)	0.6764 (69%)	0.6768	0.6752
Top Principal Comp.	10500	0.4351	0.4351 (0%)	0.4332 (63%)	0.4333	0.4324
Top two Principal Comp.		0.6802	0.6802 (0%)	0.6776 (64%)	0.6778	0.6764

More principal components can be obtained with a simple deflation method. However, it is much complicated to guarantee orthogonality. It boils down to a different harder problem.

- ✓ Our experimental evaluation is mostly numerical; we don't have detailed evaluations on real data (e.g., on population genetics data).
- ✓ How about lower-order sparse singular vectors?
- ✓ Can we come up with a convex relaxation (e.g., an PSD relaxation) and use randomized rounding to give provable bounds for the sparsity vs. accuracy tradeoff for the top (or top few) singular vectors?
- ✓ How robust is sparse PCA to input noise?

Thank you!

Questions?

Kimon Fountoulakis, Abhisek Kundu, Eugenia-Maria Kontopoulou and Petros Drineas (2016), *A Randomized Rounding Algorithm for Sparse PCA*, accepted for publication in ACM TKDD, ArXiv link.