Sparse Principal Component Analysis using Semidefinite Programming

A. d'Aspremont, L. El Ghaoui, M. Jordan, G. Lanckriet
Princeton University, U.C. Berkeley, U.C. San Diego

Support from NSF, DHS and Google.

Introduction

Principal Component Analysis (PCA)

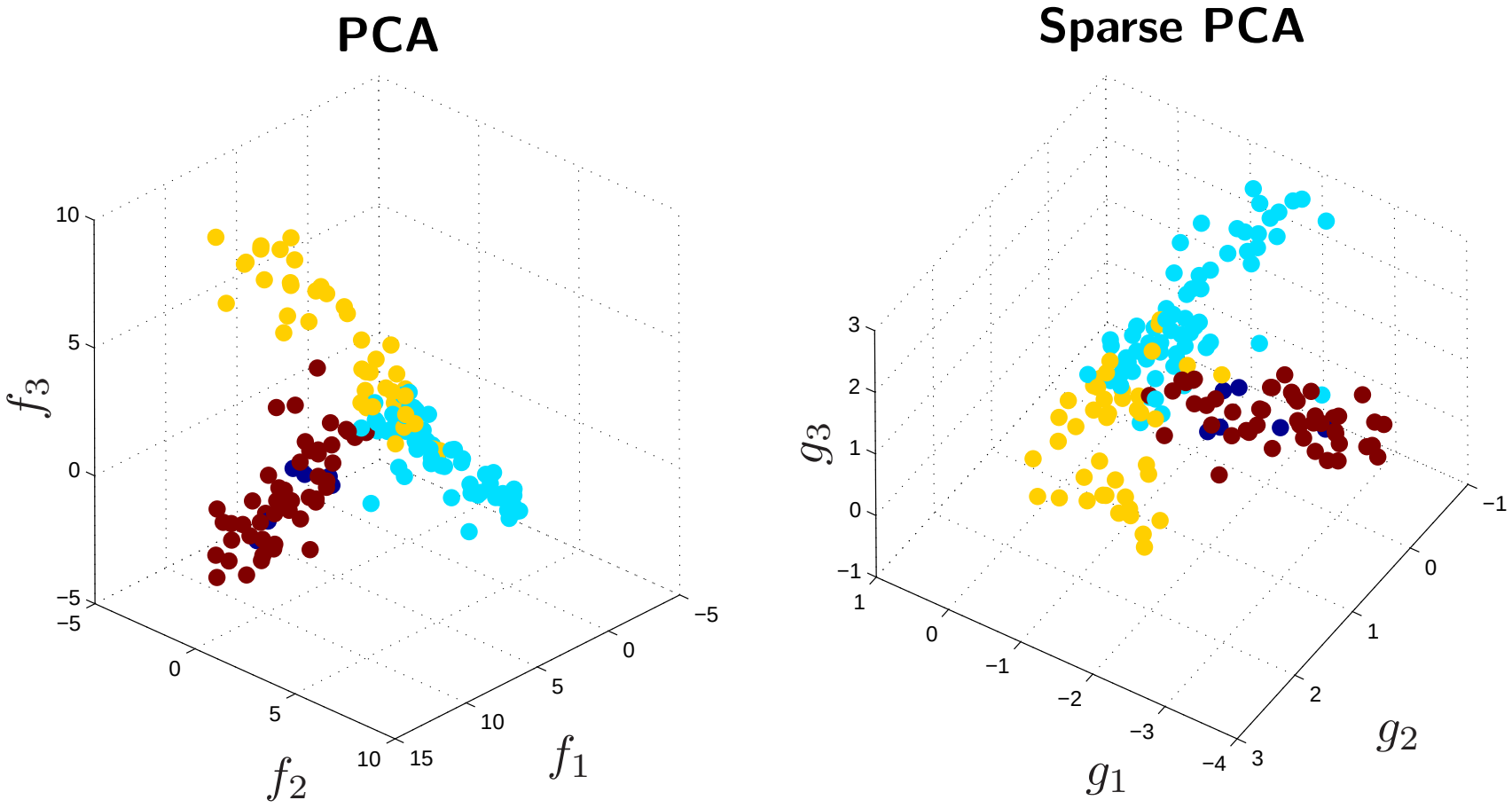
- Classical dimensionality reduction tool.
- Numerically cheap: $O(kn^2)$, only requires computing k leading eigenvectors.

Sparse PCA

- Seeks factors with a few nonzero coefficients.
- **Sparse** factors capture maximum variance and improve **interpretability**.
- Numerically hard: sparsity makes it a combinatorial problem.

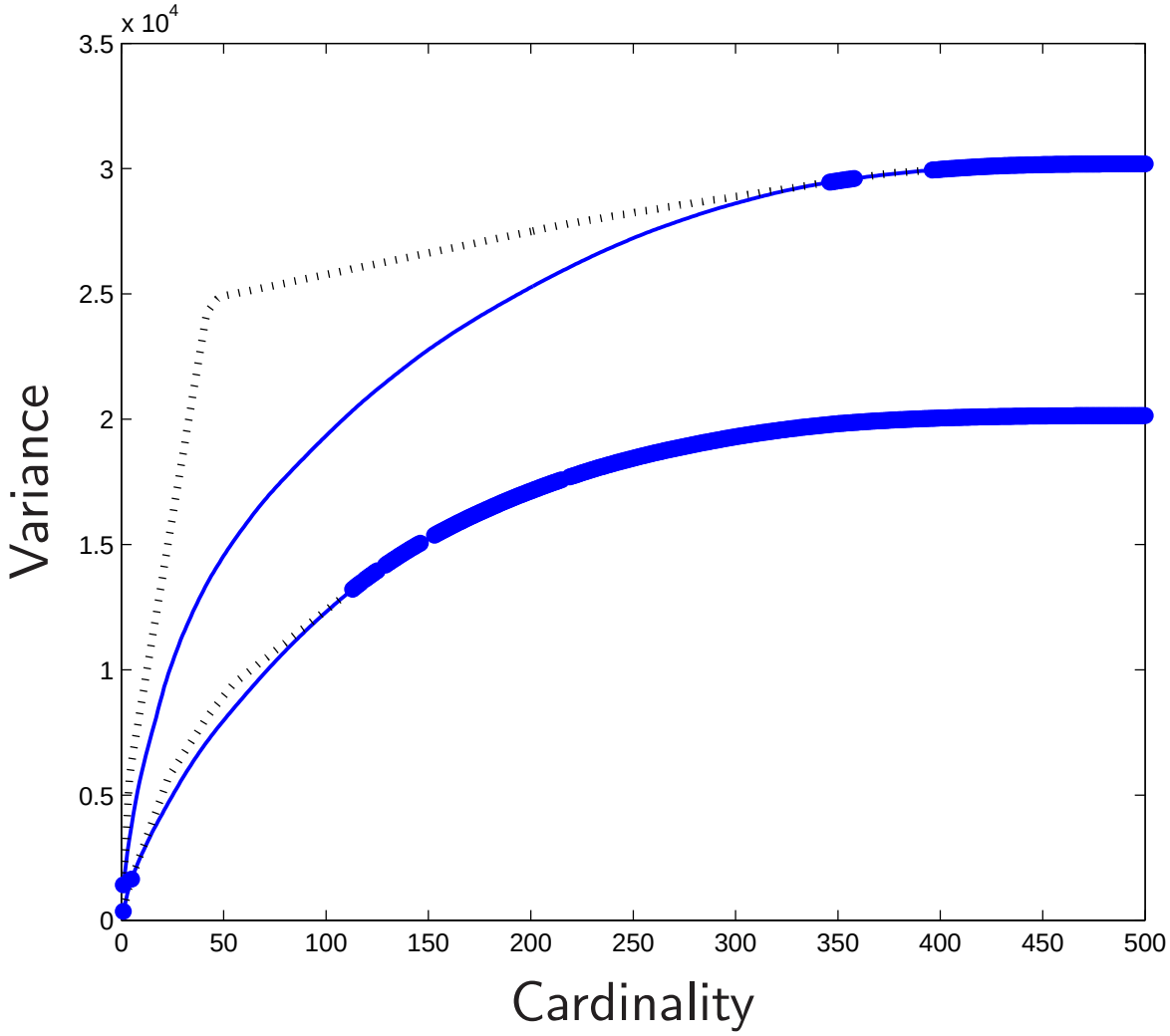
Introduction

Clustering of gene expression data in PCA versus sparse PCA, on 500 genes.



The PCA factors f_i on the left are dense and each use all 500 genes.
The sparse factors g_1, g_2 and g_3 on the right involve 6, 4 and 4 genes respectively.

Introduction



Variance (solid lines) versus cardinality tradeoff curve for two gene expression data sets, lymphoma (top) and colon cancer (bottom).

Introduction

Given a (centered) data set $A \in \mathbf{R}^{n\times m}$ composed of m observations on n variables, we form the covariance matrix $C = AA^T/(m-1)$.

Principal Component Analysis. To get the first factor, we solve:

$$\begin{array}{ll} \text{maximize} & x^TCx \\ \text{subject to} & \|x\| = 1, \end{array}$$

in the variable $x \in \mathbf{R}^n$, i.e. we maximize the **variance** explained by the **factor** x.

Sparse Principal Component Analysis. We constrain the cardinality of the factor x and solve instead:

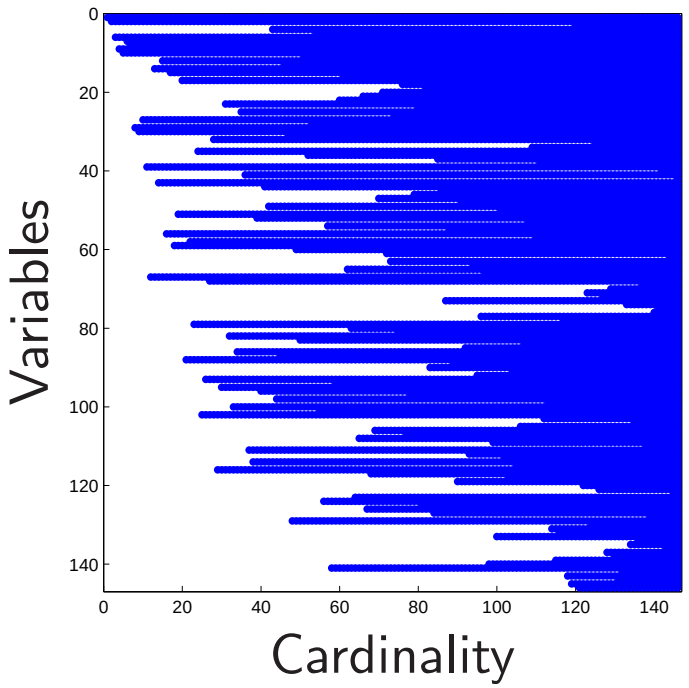
$$\begin{array}{ll} \text{maximize} & x^TCx \\ \text{subject to} & \mathbf{Card}(x) = k \\ & \|x\| = 1, \end{array}$$

in the variable $x \in \mathbf{R}^n$, where $\mathbf{Card}(x)$ is the number of nonzero coefficients in the vector x and $k > 0$ is a parameter controlling **sparsity**.

Sorting

Simplest solution: just sort variables according to variance.

Schur-Horn theorem: the diagonal of a matrix majorizes its eigenvalues so the diagonal of a matrix is a diffused vector of eigenvalues.



In this example, we selected variables according to their variance, but we ordered them according to their true ranking (computed from the optimal solution).

Related Work

- Cadima & Jolliffe (1995): the loadings with small absolute value are thresholded to zero.
- Johnstone & Lu (2004) apply this to ECG data and show consistency.
- Zou, Hastie & Tibshirani (2006), non-convex algo. (SPCA) based on a l_1 penalized representation of PCA as a regression problem.
- Non-convex methods (SCoTLASS) by Jolliffe, Trendafilov & Uddin (2003).
- A greedy search algorithm by Moghaddam, Weiss & Avidan (2006).

This talk is mostly about the results in d'Aspremont, El Ghaoui, Jordan & Lanckriet (2007). New results in:

- d'Aspremont, Bach & El Ghaoui (2007) compute a full approximate regularization path in $O(n^3)$.
- Sriperumbudur, Torres & Lanckriet (2007) apply D.C. algorithms to the penalized eigenvalue problem.

Outline

- Introduction
- Sparse PCA
  - **Semidefinite Relaxation**
  - Smooth Optimization
- Sparse Eigenvalues
  - Variable Selection
  - Compressed Sensing
- Numerical Experiments

Related Work

- Non-convex methods methods produce approximate solution with unpredictable complexity.
- Here, we produce approximate solutions with **predictable complexity**, together with bounds on suboptimality.

Combine two classic relaxation techniques:

- The lifting procedure à la MAXCUT by Goemans & Williamson (1995).
- A ℓ_1 norm relaxation of the cardinality constraint. Used in basis pursuit by Chen, Donoho & Saunders (2001), LASSO by Tibshirani (1996), etc.

Semidefinite relaxation

Start from:

$$\begin{array}{ll} \text{maximize} & x^T A x \\ \text{subject to} & \|x\|_2 = 1 \\ & \mathbf{Card}(x) \leq k, \end{array}$$

where $x \in \mathbf{R}^n$. Let $X = xx^T$ and write everything in terms of the matrix X:

$$\begin{array}{ll} \text{maximize} & \mathbf{Tr}(AX) \\ \text{subject to} & \mathbf{Tr}(X) = 1 \\ & \mathbf{Card}(X) \leq k^2 \\ & X = xx^T, \end{array}$$

Replace $X = xx^T$ by the equivalent $X \succeq 0, \ \mathbf{Rank}(X) = 1$:

$$\begin{array}{ll} \text{maximize} & \mathbf{Tr}(AX) \\ \text{subject to} & \mathbf{Tr}(X) = 1 \\ & \mathbf{Card}(X) \leq k^2 \\ & X \succeq 0, \ \mathbf{Rank}(X) = 1, \end{array}$$

again, this is the **same problem**.

Semidefinite relaxation

We have made **some progress**:

- The objective $\mathbf{Tr}(AX)$ is now **linear** in X
- The (non-convex) constraint $\|x\|_2 = 1$ became a **linear** constraint $\mathbf{Tr}(X) = 1$.

But this is still a hard problem:

- The $\mathbf{Card}(X) \leq k^2$ is still non-convex.
- So is the constraint $\mathbf{Rank}(X) = 1$.

We still need to relax the two non-convex constraints above:

- If $u \in \mathbf{R}^p$, $\mathbf{Card}(u) = q$ implies $\|u\|_1 \leq \sqrt{q}\|u\|_2$. So we can replace $\mathbf{Card}(X) \leq k^2$ by the weaker (but **convex**): $\mathbf{1}^T|X|\mathbf{1} \leq k$.
- We simply drop the rank constraint

Semidefinite Programming

Semidefinite relaxation:

$$\begin{array}{ll} \text{maximize} & x^T A x \\ \text{subject to} & \|x\|_2 = 1 \\ & \mathbf{Card}(x) \leq k, \end{array}$$

becomes

$$\begin{array}{ll} \text{maximize} & \mathbf{Tr}(AX) \\ \text{subject to} & \mathbf{Tr}(X) = 1 \\ & \mathbf{1}^T |X| \mathbf{1} \leq k \\ & X \succeq 0, \end{array}$$

- This is a **semidefinite program** in the variable $X \in \mathbf{S}^n$. . .
- Solve small problems (a few hundred variables) using IP solvers, etc.
- Dimensionality reduction apps: solve very large instances.

Solution: use first order algorithm. . .

Robustness & Tightness

Robustness. The penalized problem can be written:

$$\min_{\{|U_{ij}|\leq\rho\}} \quad \lambda^{\max}(A+U)$$

Natural interpretation: **robust** maximum eigenvalue problem with componentwise noise of magnitude ρ on the coefficients of the matrix A.

Tightness. The KKT optimality conditions are here:

$$\begin{cases} (A+U)X = \lambda^{\max}(A+U)X \\ U \circ X = \rho|X| \\ \mathbf{Tr}(X) = 1,\ X \succeq 0 \\ |U_{ij}| \leq \rho, \quad i,j = 1,\ldots,n. \end{cases}$$

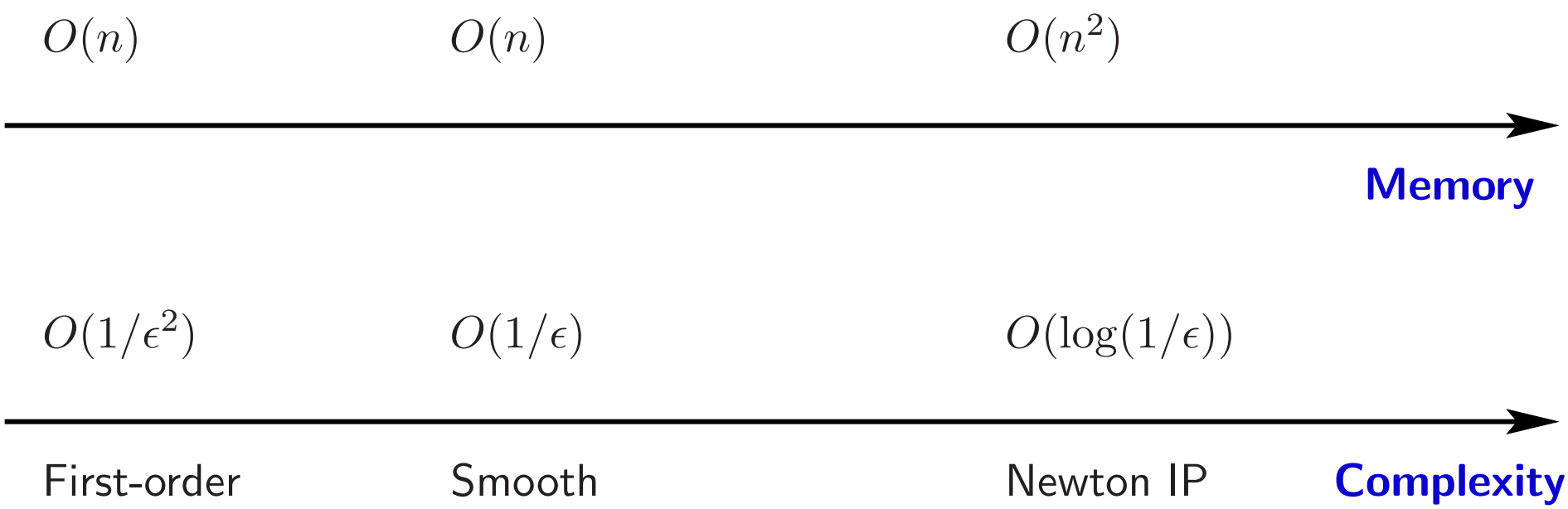
If the eigenvalue $\lambda^{\max}(A+U)$ is simple, $\mathbf{Rank}(X) = 1$ and the semidefinite relaxation is **tight**.

Outline

- Introduction
- Sparse PCA
  - Semidefinite Relaxation
  - **Smooth Optimization**
- Sparse Eigenvalues
  - Variable Selection
  - Compressed Sensing
- Numerical Experiments

First order algorithm

Complexity options. . .



First order algorithm

Here, we can exploit problem structure

- Our problem here has a particular **min-max** structure:

$$\min_{|U_{ij}|\leq\rho} \quad \max_{X\in\mathbf{S}^n} \mathbf{Tr}((A+U)X)$$

- This min-max structure means that we can use prox function algorithms by Nesterov (2005) (see also Nemirovski (2004)) to solve large, dense problem instances.

First order algorithm

If problem has min-max model, **two steps**:

- **Regularization**. Add strongly convex penalty inside the min-max representation to produce an ϵ-approximation of f with Lipschitz continuous gradient (generalized Moreau-Yosida regularization step, see Lemaréchal & Sagastizábal (1997) for example).
- **Optimal first order minimization**. Use optimal first order scheme for Lipschitz continuous functions detailed in Nesterov (1983) to the solve the regularized problem.

Benefits:

- Produces an ϵ solution is given by $O\left(1/\epsilon\right)$ compared to $O\left(1/\epsilon^2\right)$ for generic first-order methods.
- Low memory requirements. Change in **granularity** of the solver: larger number of cheaper iterations.

Caveat: Two (projection) subproblems need to be solved very efficiently. . .

First order algorithm

Regularization. Let $\mu > 0$ and $X \in \mathbf{S}_n$, we define:

$$f_\mu(X) = \mu \log \mathbf{Tr}\left(\exp\left(\frac{X}{\mu}\right)\right)$$

We then have:

$$\lambda^{\max}(X) \leq f_\mu(X) \leq \lambda^{\max}(X) + \mu \log n,$$

so if we set $\mu = \epsilon / \log n$, this becomes a **uniform ϵ-approximation** of $\lambda^{\max}(X)$ with a **Lipschitz continuous gradient** with constant:

$$L = \frac{1}{\mu} = \frac{\log n}{\epsilon}.$$

The gradient $\nabla f_\mu(X)$ can be computed explicitly in $O(n^3)$ as:

$$\exp\left(\frac{X - \lambda^{\max}(X)\mathbf{I}}{\mu}\right) / \mathbf{Tr}\left(\exp\left(\frac{X - \lambda^{\max}(X)\mathbf{I}}{\mu}\right)\right)$$

using the same matrix exponential.

First order algorithm

Optimal first-order minimization. The minimization algorithm in Nesterov (1983) then involves the following steps:

Choose $\epsilon > 0$ and set $X_0 = \beta I_n$, **For** $k = 0, \ldots, N$ **do**

1. Compute $\nabla f_\epsilon(X_k)$
2. Find $Y_k = \arg\min_{Y \in \mathcal{Q}} \{\mathbf{Tr}(\nabla f_\epsilon(X_k)(Y - X_k)) + \frac{1}{2} L_\epsilon \|Y - X_k\|_F^2\}$.
3. Find $Z_k = \arg\min_{X \in \mathcal{Q}} \left\{ L_\epsilon \beta^2 d_1(X) + \sum_{i=0}^{k} \frac{i+1}{2} \mathbf{Tr}(\nabla f_\epsilon(X_i)(X - X_i)) \right\}$.
4. Update $X_k = \frac{2}{k+3} Z_k + \frac{k+1}{k+3} Y_k$.
5. Test if gap less than target precision.

- **Step 1** requires computing a matrix exponential.
- **Steps 2 and 3** are both Euclidean projections on $\mathcal{Q} = \{U \in \mathbf{S}^n : \ |U_{ij}| \leq \rho\}$.

First order algorithm

Complexity:

- The number of iterations to get accuracy ϵ is

$$O\left(\frac{n\sqrt{\log n}}{\epsilon}\right)$$

- At each iteration, the cost of computing a matrix exponential up to machine precision is $O(n^3)$.

Computing matrix exponentials:

- Many options, cf. "Nineteen Dubious Ways to Compute the Exponential of a Matrix" by Moler & Van Loan (1978), Moler & Van Loan (2003).
- Padé approximation, full eigenvalue decomposition: $O(n^3)$ up to machine precision.
- In practice, machine precision is unnecessary and a partial eigenvalue decomposition is enough (see d'Aspremont (2005)).

Outline

- Introduction
- Sparse PCA
  - Semidefinite Relaxation
  - Smooth Optimization
- **Sparse Eigenvalues**
  - Variable Selection
  - Compressed Sensing
- Numerical Experiments

Sparse Eigenvalues

Combining semidefinite and ℓ_1 relaxations, we obtained:

$$\begin{array}{rll} \lambda^k_{\max}(A) \leq & \text{max.} & \mathbf{Tr}(AX) \\ & \text{s.t.} & \mathbf{Tr}(X) = 1 \\ & & \mathbf{1}^T|X|\mathbf{1} \leq k \\ & & X \succeq 0, \end{array}$$

This relaxation produces **upper bounds** on sparse (or restricted) maximum eigenvalues. Similarly, we can get lower bounds on sparse minimum eigenvalues.

- Used to bound MSE and model consistency in LASSO (sparse least-squares).
- Control recovery rates in compressed sensing.

LASSO

Assume that observations $(Y_1, \ldots, Y_n)$ follow a linear model:

$$Y = X\beta + \epsilon$$

where $\beta \in \mathbf{R}^p$ and $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. We define the LASSO estimator of β as:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \|Y - X\beta\|_2^2 + \lambda\|\beta\|_1.$$

Consistency.

- Suppose b is **sparse** with cardinality $s(n)$, Meinshausen & Yu (2007) show that with probability tending to 1 as $n \to \infty$:

$$\|\beta - \hat{\beta}\|_2^2 \leq M \frac{s(n) \log p(n)}{n \lambda_{\min}^{s(n)}(X^T X)}$$

- Meinshausen & Yu (2007) also show sign consitency based on sparse eigenvalues. Similar non-asymptotic result by Candès & Tao (2007).

Compressed Sensing

Following Candès & Tao (2005) (see also Donoho & Tanner (2005)), recover a signal $f \in \mathbf{R}^n$ from corrupted measurements y:

$$y = Af + e,$$

where $A \in \mathbf{R}^{m \times n}$ is a coding matrix and $e \in \mathbf{R}^m$ is an unknown **sparse** vector of errors.

This amounts to solving the following (combinatorial) problem:

$$\begin{array}{ll} \text{minimize} & \mathbf{Card}(x) \\ \text{subject to} & Fx = Fy \end{array}$$

where $F \in \mathbf{R}^{p \times m}$ is a matrix such that $FA = 0$.

Compressed Sensing: Restricted Isometry Constant

Given a matrix $F \in \mathbf{R}^{p \times m}$ and $0 < S \leq m$, its **restricted isometry** constant δ_S is the smallest number such that for any subset $I \subset [1, m]$ of cardinality at most S we have:

$$(1 - \delta_S)\|c\|^2 \leq \|F_I c\|^2 \leq (1 + \delta_S)\|c\|^2,$$

for all $c \in \mathbf{R}^{|I|}$, where F_I is the submatrix of F formed by keeping only the columns of F in the set I.

Compressed sensing: perfect recovery

The following result then holds.

Proposition 1. ***Candès & Tao (2005)***. *Suppose that the restricted isometry constants of a matrix* $F \in \mathbf{R}^{p \times m}$ *satisfy :*

$$\delta_S + \delta_{2S} + \delta_{3S} < 1 \tag{1}$$

for some integer S *such that* $0 < S \leq m$*, then if* x *is an optimal solution of the convex program:*

$$\begin{array}{ll} \textit{minimize} & \|x\|_1 \\ \textit{subject to} & Fx = Fy \end{array}$$

such that $\mathbf{Card}(x) \leq S$ *then* x *is also an optimal solution of the combinatorial problem:*

$$\begin{array}{ll} \textit{minimize} & \mathbf{Card}(x) \\ \textit{subject to} & Fx = Fy. \end{array}$$

Compressed sensing: restricted isometry

The restricted isometry constant δ_S in condition can be computed by solving the following sparse PCA problem:

$$\begin{array}{rll} (1+\delta_S) = & \text{max.} & x^T(F^TF)x \\ & \text{s. t.} & \mathbf{Card}(x) \leq S \\ & & \|x\| = 1, \end{array}$$

in the variable $x \in \mathbf{R}^m$ (a similar sparse PCA problem gives the other inequality).

- Candès & Tao (2005) obtain an **asymptotic** proof that some random matrices satisfy the restricted isometry condition with **overwhelming probability** (i.e. exponentially small probability of failure)
- Upper bounds for sparse PCA prove **deterministically** and with **polynomial complexity** that a finite dimensional matrix satisfies the restricted isometry condition.

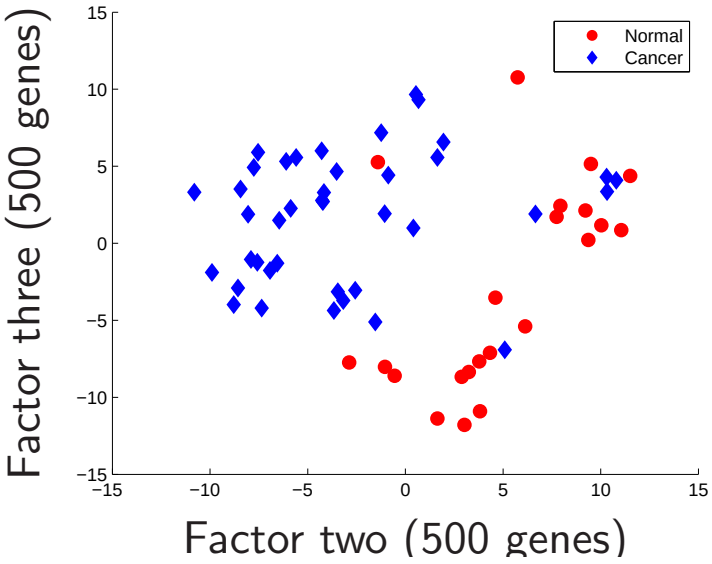
Outline

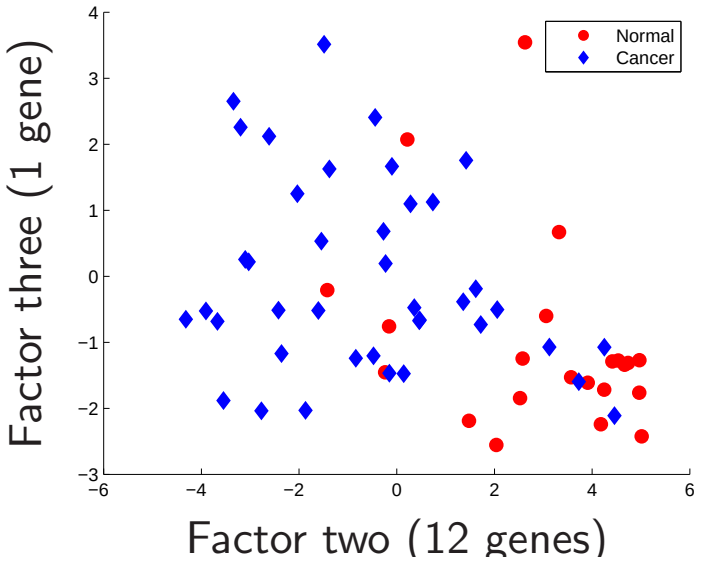
- Introduction
- Sparse PCA
  - Semidefinite Relaxation
  - Smooth Optimization
- Sparse Eigenvalues
  - Variable Selection
  - Compressed Sensing
- **Numerical Experiments**

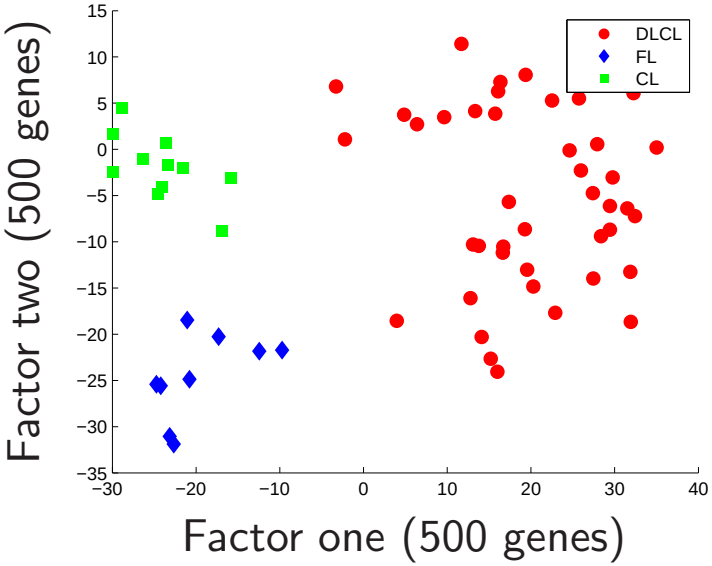
Gene Expression Data

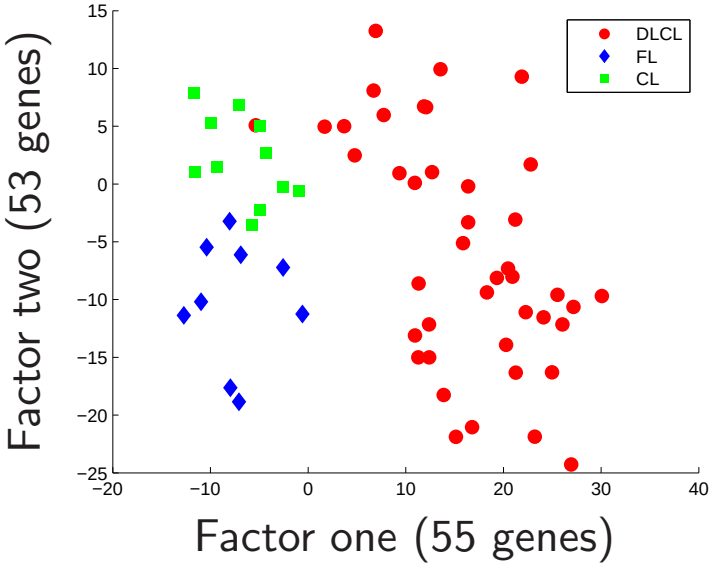
- Use sparse PCA as a crude clustering/variable selection tool (see Luss & d'Aspremont (2007)).
- Use colon cancer data set of Alon, Barkai, Notterman, Gish, Ybarra, Mack & Levine (1999), lymphoma data from Alizadeh, Eisen, Davis, Ma, Lossos & Rosenwald (2000).
- Track clustering quality versus number of genes used.

Sparse PCA: clustering



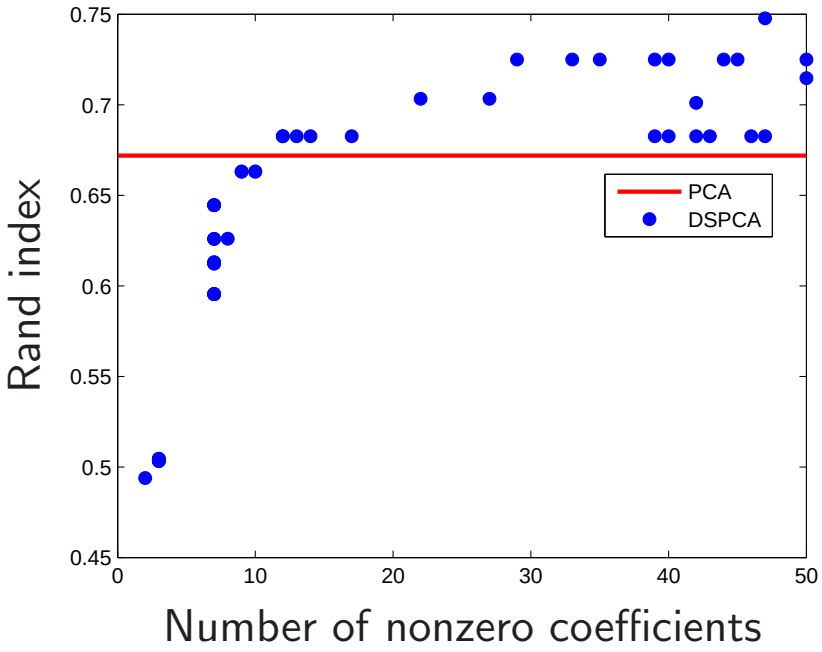


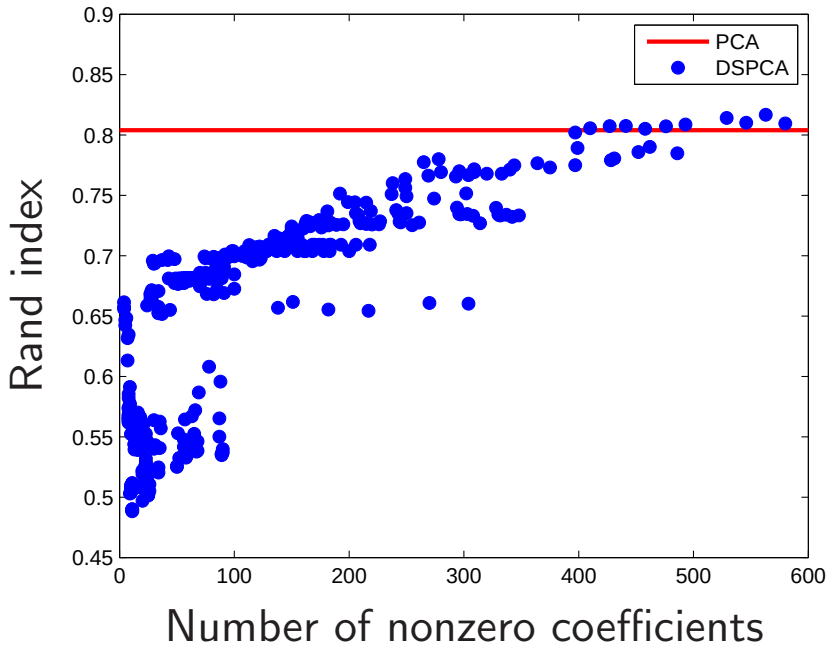




PCA (left) and DSPCA (right), colon cancer (top) and lymphoma (bottom).

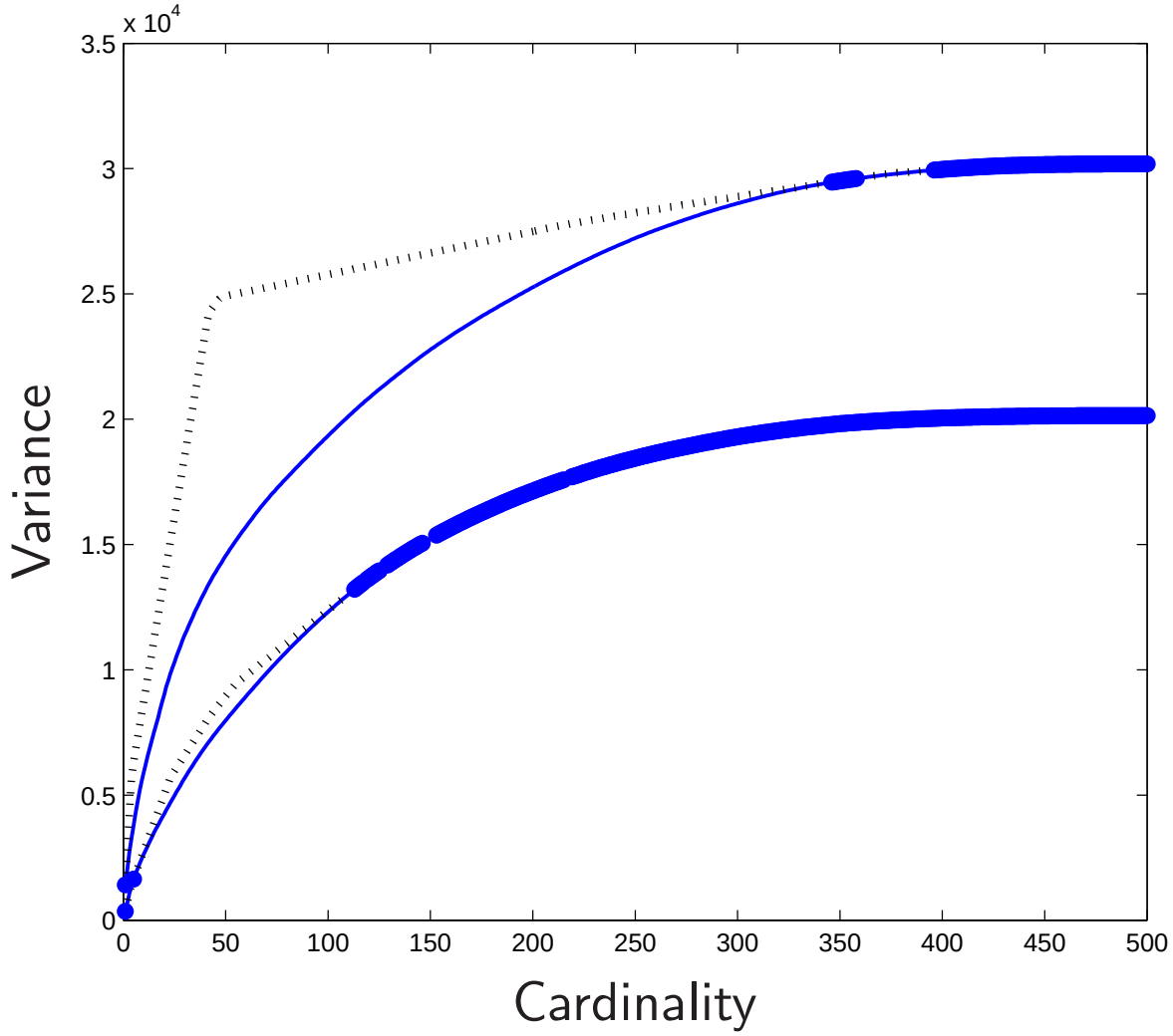
Sparse PCA: clustering





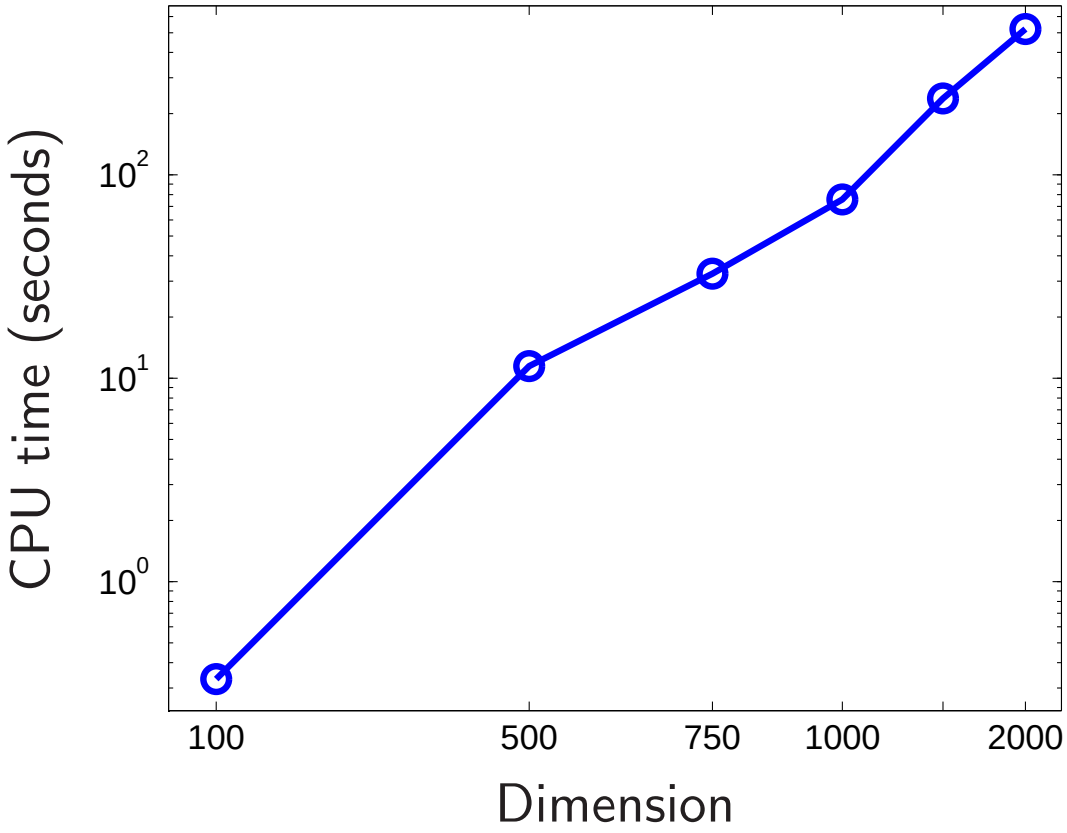
Rand index (clustering) versus sparsity: colon cancer (left) & lymphoma (right).

Tradeoff



Variance (solid lines) versus cardinality tradeoff curve for two gene expression data sets, lymphoma (top) and colon cancer (bottom).

CPU time



n	CPU time (secs)
100	0 m 1 s
500	0 m 11 s
750	1 m 33 s
1000	1 m 16 s
1500	4 m 57 s
2000	9 m 41 s

Using the data in Alon et al. (1999), with $\rho = 1$, we plot CPU time to get a 10^2 decrease in duality gap.

Conclusion

- The tradeoff between sparsity and explained variance is often favorable.
- Dense semidefinite programs solved efficiently for matrices with $n \sim 10^3$

- Slides online.
- Source code, binaries and test data available at:

`www.princeton.edu/~aspremon/DSPCA.htm`

- More results in d'Aspremont, Bach & El Ghaoui (2007) and Sriperumbudur et al. (2007).

References

Alizadeh, A., Eisen, M., Davis, R., Ma, C., Lossos, I. & Rosenwald, A. (2000), 'Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling', *Nature* **403**, 503–511.

Alon, A., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D. & Levine, A. J. (1999), 'Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays', *Cell Biology* **96**, 6745–6750.

Cadima, J. & Jolliffe, I. T. (1995), 'Loadings and correlations in the interpretation of principal components', *Journal of Applied Statistics* **22**, 203–214.

Candès, E. J. & Tao, T. (2005), 'Decoding by linear programming', *Information Theory, IEEE Transactions on* **51**(12), 4203–4215.

Candès, E. & Tao, T. (2007), 'The Dantzig selector: statistical estimation when$ p$ is much larger than$ n$', *To appear in Annals of Statistics* .

Chen, S., Donoho, D. & Saunders, M. (2001), 'Atomic decomposition by basis pursuit.', *SIAM Review* **43**(1), 129–159.

d'Aspremont, A. (2005), 'Smooth optimization with approximate gradient', *ArXiv: math.OC/0512344* .

d'Aspremont, A., Bach, F. R. & El Ghaoui, L. (2007), Full regularization path for sparse principal component analysis, *in* 'Proceedings of the 24th international conference on Machine learning', pp. 177–184.

d'Aspremont, A., El Ghaoui, L., Jordan, M. & Lanckriet, G. R. G. (2007), 'A direct formulation for sparse PCA using semidefinite programming', *SIAM Review* **49**(3), 434–448.

Donoho, D. L. & Tanner, J. (2005), 'Sparse nonnegative solutions of underdetermined linear equations by linear programming', *Proc. of the National Academy of Sciences* **102**(27), 9446–9451.

Goemans, M. & Williamson, D. (1995), 'Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming', *J. ACM* **42**, 1115–1145.

Johnstone, I. & Lu, A. Y. (2004), 'Sparse principal components analysis', *Working Paper, Stanford department of statistics* .

Jolliffe, I. T., Trendafilov, N. & Uddin, M. (2003), 'A modified principal component technique based on the LASSO', *Journal of Computational and Graphical Statistics* **12**, 531–547.

Lemaréchal, C. & Sagastizábal, C. (1997), 'Practical aspects of the Moreau-Yosida regularization: theoretical preliminaries', *SIAM Journal on Optimization* **7**(2), 367–385.

Luss, R. & d'Aspremont, A. (2007), 'Clustering and feature selection using sparse principal component analysis', *Working Paper* .

Meinshausen, N. & Yu, B. (2007), Lasso-type recovery of sparse representations for highdimensional data, Technical report, To appear in Annals of Statistics.

Moghaddam, B., Weiss, Y. & Avidan, S. (2006), 'Spectral bounds for sparse PCA: Exact and greedy algorithms', *Advances in Neural Information Processing Systems* **18**.

Moler, C. & Van Loan, C. (1978), 'Nineteen dubious ways to computethe exponentialof a matrix', *SIAM Review* **20**, 801–836.

Moler, C. & Van Loan, C. (2003), 'Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later', *SIAM Review* **45**(1), 3–49.

Nemirovski, A. (2004), 'Prox-method with rate of convergence O(1/T) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems', *SIAM Journal on Optimization* **15**(1), 229–251.

Nesterov, Y. (1983), 'A method of solving a convex programming problem with convergence rate $O(1/k^2)$', *Soviet Mathematics Doklady* **27**(2), 372–376.

Nesterov, Y. (2005), 'Smooth minimization of non-smooth functions', *Mathematical Programming* **103**(1), 127–152.

Sriperumbudur, B., Torres, D. & Lanckriet, G. (2007), 'Sparse eigen methods by DC programming', *Proceedings of the 24th international conference on Machine learning* pp. 831–838.

Tibshirani, R. (1996), 'Regression shrinkage and selection via the LASSO', *Journal of the Royal statistical society, series B* **58**(1), 267–288.

Zou, H., Hastie, T. & Tibshirani, R. (2006), 'Sparse Principal Component Analysis', *Journal of Computational & Graphical Statistics* **15**(2), 265–286.