Full Regularization Path
for Sparse Principal Component Analysis

Alexandre d'Aspremont, Francis Bach & Laurent El Ghaoui,

Princeton University, INRIA/ENS Ulm & U.C. Berkeley

Support from NSF, DHS and Google.

Introduction

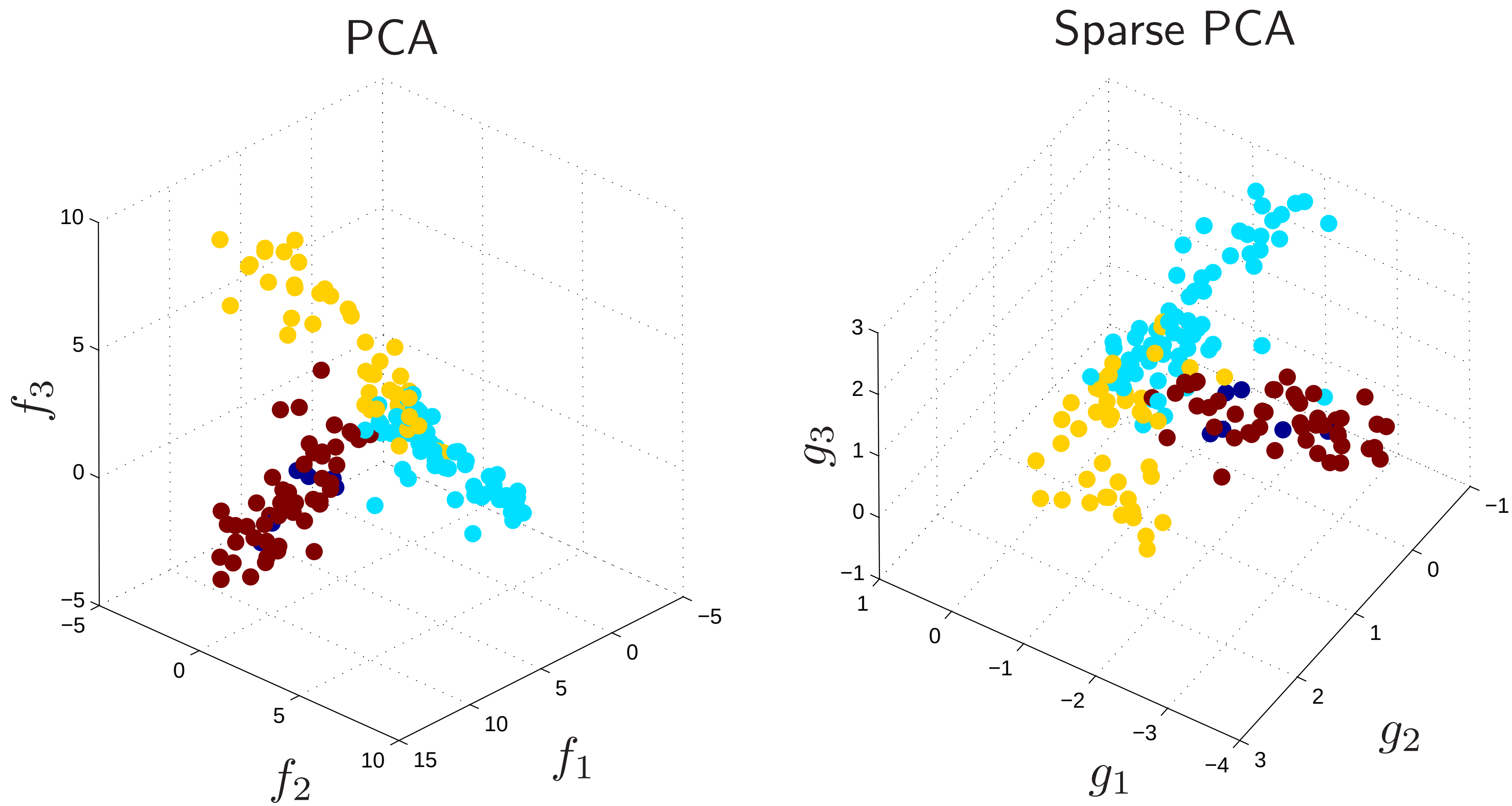
Principal Component Analysis

- Classic dimensionality reduction tool.
- Numerically cheap: $O(n^2)$ as it only requires computing a few dominant eigenvectors.

Sparse PCA

- Get **sparse** factors capturing a maximum of variance.
- Numerically hard: combinatorial problem.
- Controlling the sparsity of the solution is also hard in practice.

Introduction



Clustering of the gene expression data in the PCA versus sparse PCA basis with 500 genes. The factors f on the left are dense and each use all 500 genes while the sparse factors $g_1,\ g_2$ and g_3 on the right involve 6, 4 and 4 genes respectively. (Data: Iconix Pharmaceuticals)

Introduction

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

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

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:

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

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**.

Outline

- Introduction
- **Algorithms**
- Optimality
- Numerical Results

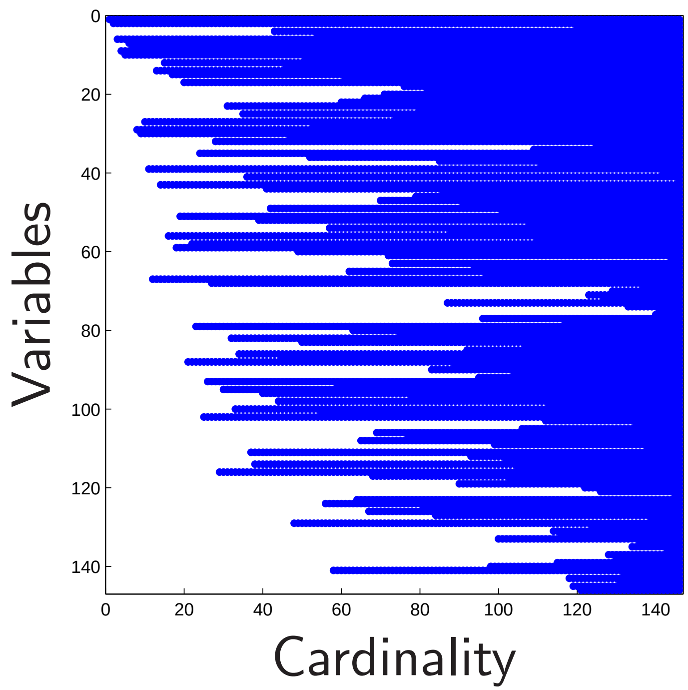
Algorithms

Existing methods. . .

- Cadima & Jolliffe (1995): the loadings with small absolute value are thresholded to zero.
- SPCA Zou, Hastie & Tibshirani (2004), non-convex algo. based on a l_1 penalized representation of PCA as a regression problem.
- A convex relaxation in d'Aspremont, El Ghaoui, Jordan & Lanckriet (2007).
- Non-convex optimization methods: SCoTLASS by Jolliffe, Trendafilov & Uddin (2003) or Sriperumbudur, Torres & Lanckriet (2007).
- A greedy algorithm by Moghaddam, Weiss & Avidan (2006*b*).

Algorithms

Simplest solution: just sort variables according to variance, keep the k variables with highest variance. **Schur-Horn theorem**: the diagonal of a matrix majorizes its eigenvalues.



Other simple solution: **Thresholding**, compute the first factor x from regular PCA and keep the k variables corresponding to the k largest coefficients.

Algorithms

Greedy search (see Moghaddam et al. (2006*b*)). Written on the square root here.

1. Preprocessing. Permute elements of Σ accordingly so that its diagonal is decreasing. Compute the Cholesky decomposition $\Sigma = A^T A$. Initializate $I_1 = \{1\}$ and $x_1 = a_1 / \|a_1\|$.

2. Compute
$$i_k = \underset{i \notin I_k}{\operatorname{argmax}} \lambda_{max} \left(\sum_{j \in I_k \cup \{i\}} a_j a_j^T \right)$$

3. Set $I_{k+1} = I_k \cup \{i_k\}$.

4. Compute x_{k+1} as the dominant eigenvector of $\sum_{j \in I_{k+1}} a_j a_j^T$.

5. Set $k = k + 1$. If $k < n$ go back to step 2.

Algorithms: complexity

Greedy Search

- Iteration k of the greedy search requires computing $(n-k)$ maximum eigenvalues, hence has complexity $O((n-k)k^2)$ if we exploit the Gram structure.
- This means that computing a full path of solutions has complexity $O(n^4)$.

Approximate Greedy Search

- We can exploit the following first-order inequality:

$$\lambda_{max}\left(\sum_{j\in I_k\cup\{i\}} a_j a_j^T\right) \geq \lambda_{max}\left(\sum_{j\in I_k} a_j a_j^T\right) + (a_i^T x_k)^2$$

  where x_k is the dominant eigenvector of $\sum_{j\in I_k} a_j a_j^T$.
- We only need to solve one maximum eigenvalue problem per iteration, with cost $O(k^2)$. The complexity of computing a full path of solution is now $O(n^3)$.

Algorithms

Approximate greedy search.

1. Preprocessing. Permute elements of Σ accordingly so that its diagonal is decreasing. Compute the Cholesky decomposition $\Sigma = A^T A$. Initializate $I_1 = \{1\}$ and $x_1 = a_1/\|a_1\|$.

2. Compute $i_k = \mathrm{argmax}_{i \notin I_k} (x_k^T a_i)^2$

3. Set $I_{k+1} = I_k \cup \{i_k\}$.

4. Compute x_{k+1} as the dominant eigenvector of $\sum_{j \in I_{k+1}} a_j a_j^T$.

5. Set $k = k + 1$. If $k < n$ go back to step 2.

Outline

- Introduction
- Algorithms
- **Optimality**
- Numerical Results

Algorithms: optimality

- We can write the sparse PCA problem in penalized form:

$$\max_{\|x\| \leq 1} x^T C x - \rho \, \mathbf{Card}(x)$$

in the variable $x \in \mathbf{R}^n$, where $\rho > 0$ is a parameter controlling sparsity.

- This problem is equivalent to solving:

$$\max_{\|z\|=1} \sum_{i=1}^{n} ((a_i^T z)^2 - \rho)_+$$

in the variable $x \in \mathbf{R}^n$, where the matrix A is the Cholesky decomposition of C, with $C = A^T A$. We only keep variables for which $(a_i^T z)^2 \geq \rho$.

Algorithms: optimality

The problem

$$\max_{\|z\|=1} \sum_{i=1}^{n} ((a_i^T z)^2 - \rho)_+$$

is a convex maximization problem, hence is still hard. We can formulate a semidefinite relaxation by writing it in the equivalent form:

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^{n} \mathbf{Tr}(X^{1/2} a_i a_i^T X^{1/2} - \rho X)_+ \\ \text{subject to} & \mathbf{Tr}(X) = 1,\ X \succeq 0,\ \mathbf{Rank}(X) = 1, \end{array}$$

in the variable $X \in \mathbf{S}_n$ with $X = zz^T$. If we drop the rank constraint, this becomes a convex problem and using

$$\mathbf{Tr}(X^{1/2} B X^{1/2})_+ = \min_{\{Y \succeq B,\ Y \succeq 0\}} \mathbf{Tr}(YX),$$

we can get its dual as:

$$\begin{array}{ll} \text{max.} & \sum_{i=1}^{n} \mathbf{Tr}(P_i B_i) \\ \text{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0,\ X \succeq P_i \succeq 0, \end{array}$$

which is a semidefinite program in the variables $X \in \mathbf{S}_n,\ P_i \in \mathbf{S}_n$.

Algorithms: optimality

- When the solution of this last SDP has rank one, it also produces a globally optimal solution for the sparse PCA problem.
- In practice, this semidefinite program but we can use it to test the optimality of the solutions computed by the approximate greedy method.
- When the SDP has a rank one, the KKT optimality conditions for the semidefinite relaxation are given by:

$$\left\{ \begin{array}{l} \left(\sum_{i=1}^n Y_i\right) X = \lambda_{\max}\left(\sum_{i=1}^n Y_i\right) X \\ x^T Y_i x = \left\{ \begin{array}{l} (a_i^T x)^2 - \rho \text{ if } i \in I \\ 0 \text{ if } i \in I^c \end{array} \right. \\ Y_i \succeq B_i, \ Y_i \succeq 0. \end{array} \right.$$

- This is a (large) semidefinite feasibility problem, but a good guess for Y_i often turns out to be sufficient.

Algorithms: optimality

Optimality: sufficient conditions. Given a sparsity pattern I, setting x to be the largest eigenvector of $\sum_{i \in I} a_i a_i^T$. If there is a parameter ρ_I such that:

$$\max_{i \notin I} (a_i^T x)^2 \leq \rho_I \leq \min_{i \in I} (a_i^T x)^2.$$

and

$$\begin{pmatrix} \mathbf{diag}((x^T a_i)^2) - \rho_I \mathbf{I} & (a_i a_i^T x)_{i \in I}^T - \rho_I \mathbf{1} x^T \\ (a_i a_i^T x)_{i \in I} - \rho_I x \mathbf{1}^T & \sum_i (x^T a_i)^2 \mathbf{I} - \rho_I \mathbf{I} \end{pmatrix} \succeq 0$$

with

$$\lambda_{\max} \left(\sum_{i \in I} \frac{B_i x x^T B_i}{x^T B_i x} + \sum_{i \in I^c} Y_i \right) \leq \sigma$$

where

$$Y_i = \max \left\{ 0, \rho \frac{(a_i^T a_i - \rho)}{(\rho - (a_i^T x)^2)} \right\} \frac{(\mathbf{I} - x x^T) a_i a_i^T (\mathbf{I} - x x^T)}{\|(\mathbf{I} - x x^T) a_i\|^2}, \quad i \in I^c.$$

Then the vector z such that $z = \operatorname{argmax}_{\{z_{I^c}=0,\ \|z\|=1\}} z^T \Sigma z$, which is formed by padding zeros to the dominant eigenvector of the submatrix $\Sigma_{I,I}$ is a global solution to the sparse PCA problem for $\rho = \rho_I$.

Optimality: why bother?

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

$$y = Af + e,$$

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

This is equivalent to solving the following (combinatorial) problem:

$$\begin{array}{ll} \text{minimize} & \|x\|_0 \\ \text{subject to} & Fx = Fy \end{array}$$

where $\|x\|_0 = \mathbf{Card}(x)$ and $F \in \mathbf{R}^{p \times m}$ is a matrix such that $FA = 0$.

Compressed sensing: restricted isometry

Candès & Tao (2005): given a matrix $F \in \mathbf{R}^{p \times m}$ and an integer S such that $0 < S \leq m$, we define its **restricted isometry** constant δ_S as 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 :*

$$\delta_S + \delta_{2S} + \delta_{3S} < 1/4 \tag{1}$$

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:

$$\begin{array}{ll} \textit{minimize} & \|x\|_0 \\ \textit{subject to} & Fx = Fy. \end{array}$$

Compressed sensing: restricted isometry

The restricted isometry constant δ_S in condition (1) 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 $x \in \mathbf{R}^m$ and another sparse PCA problem on $\alpha\mathbf{I} - F^TF$ to get 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)
- When they hold, the optimality conditions and upper bounds for sparse PCA allow us to prove (**deterministically** and with **polynomial complexity**) that a finite dimensional matrix satisfies the restricted isometry condition.

Outline

- Introduction
- Algorithms
- Optimality
- **Numerical Results**

Numerical Results

Artficial data. We generate a matrix U of size 150 with uniformly distributed coefficients in $[0,1]$. We let $v \in \mathbf{R}^{150}$ be a sparse vector with:

$$v_i = \begin{cases} 1 & \text{if } i \leq 50 \\ 1/(i-50) & \text{if } 50 < i \leq 100 \\ 0 & \text{otherwise} \end{cases}$$

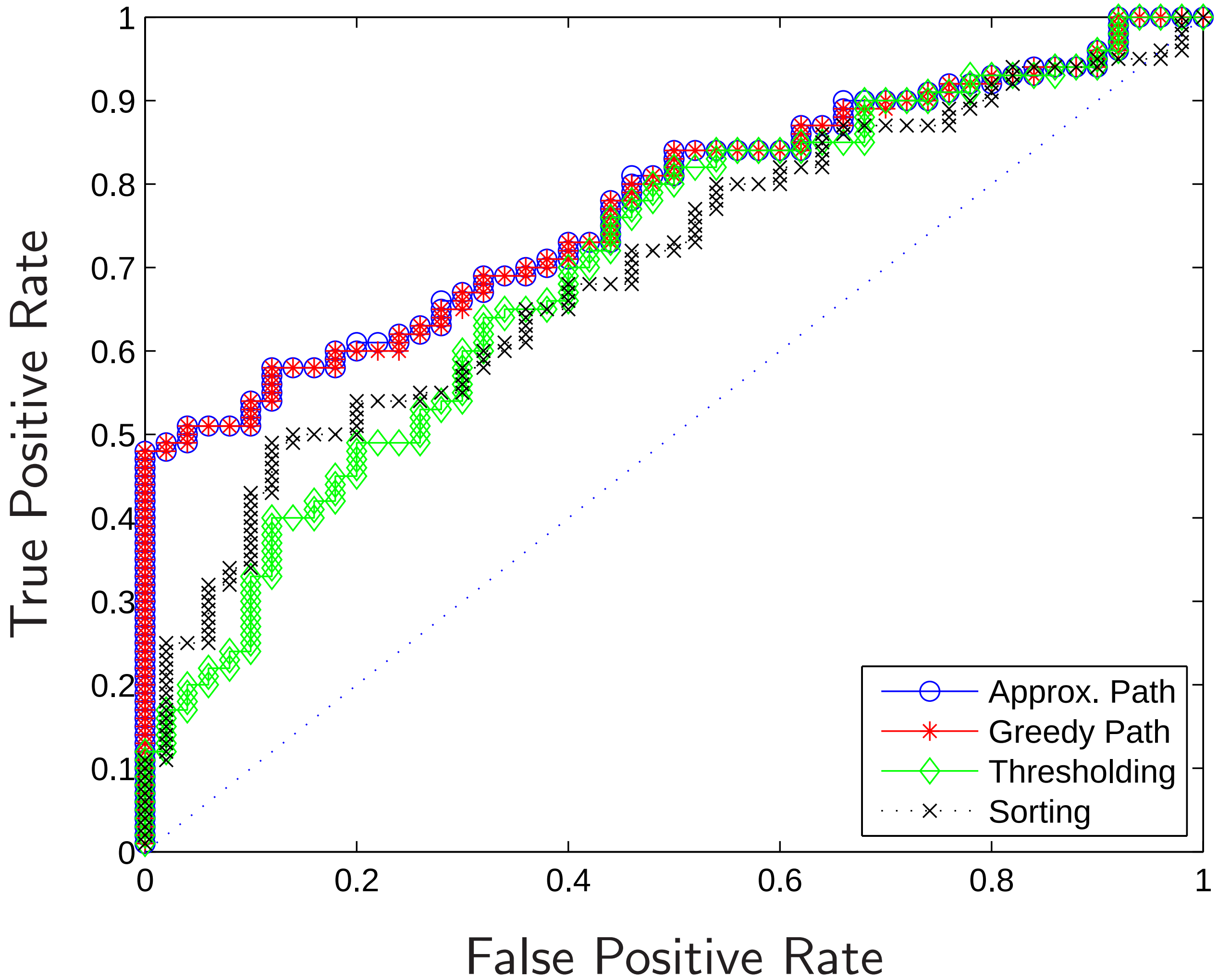
We form a test matrix

$$\Sigma = U^T U + \sigma v v^T,$$

where σ is the signal-to-noise ratio.

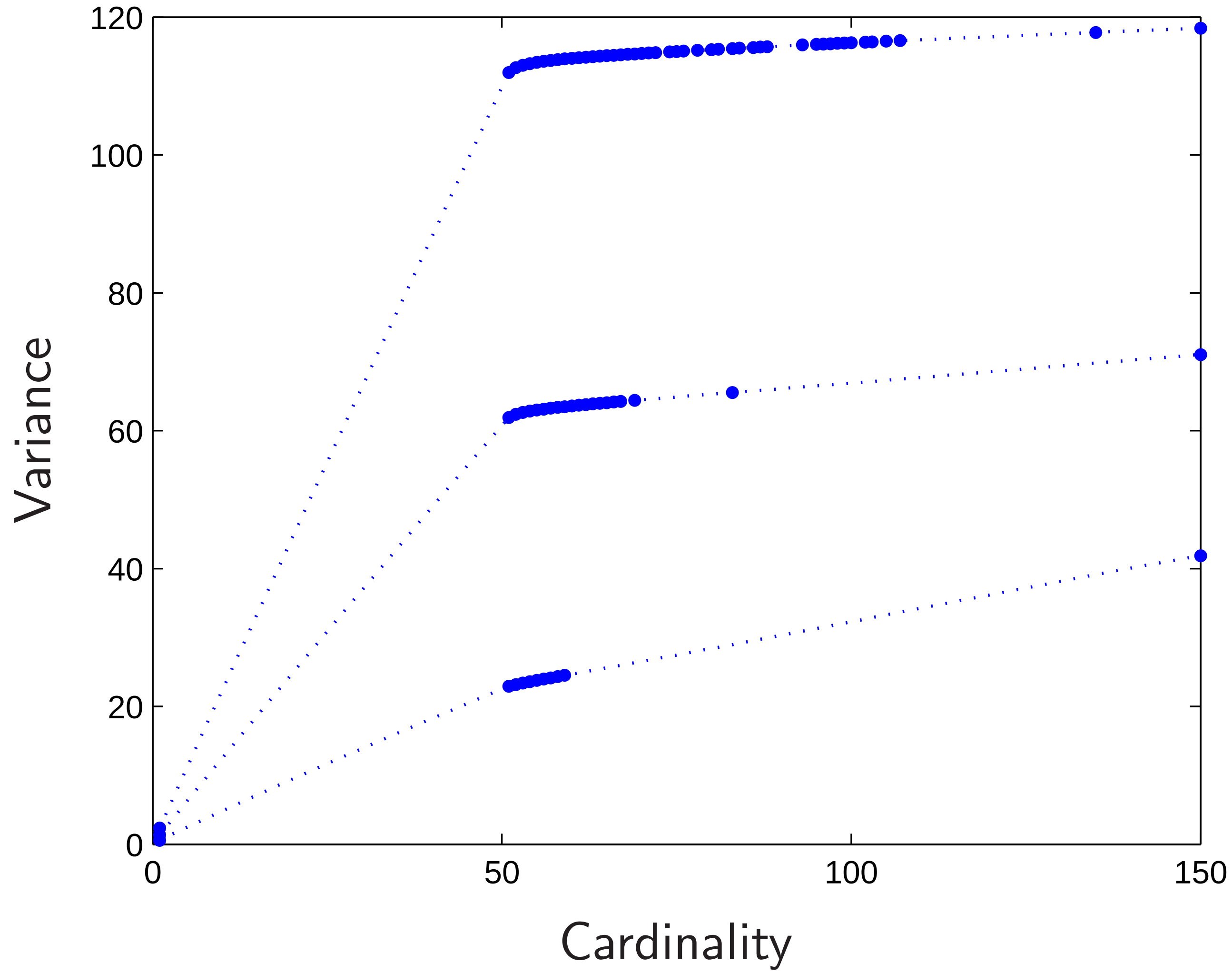
Gene expression data. We run the approximate greedy algorithm on two gene expression data sets, one on **colon cancer** from Alon, Barkai, Notterman, Gish, Ybarra, Mack & Levine (1999), the other on **lymphoma** from Alizadeh, Eisen, Davis, Ma, Lossos & Rosenwald (2000). We only keep the 500 genes with largest variance.

Numerical Results



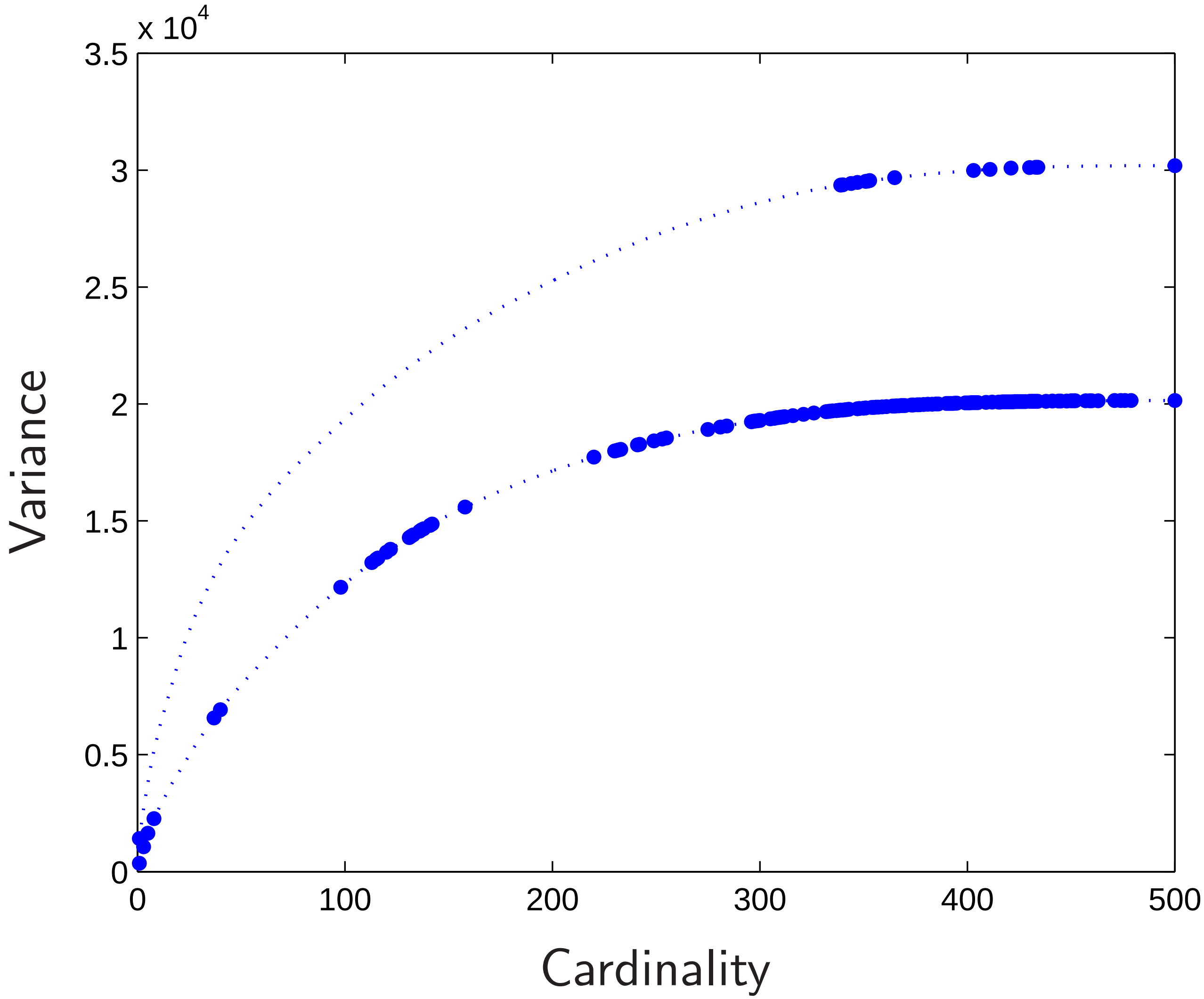
ROC curves for sorting, thresholding, fully greedy solutions and approximate greedy solutions for $\sigma = 2$.

Numerical Results



Variance versus cardinality tradeoff curves for $\sigma = 10$ (bottom), $\sigma = 50$ and $\sigma = 100$ (top). Optimal points are in bold.

Numerical Results



Variance versus cardinality tradeoff curve for two gene expression data sets, lymphoma (top) and colon cancer (bottom). Optimal points are in bold.

Conclusion & Extensions

Sparse PCA in **practice**, if your problem has. . .

- A **million** variables: can't even form a covariance matrix. **Sort** variables according to variance and keep a few thousand.
- A few **thousand** variables (more if Gram format): **approximate greedy** method described here.
- A few **hundred** variables: use DSPCA, SPCA, **full greedy** search, etc.

Of course, these techniques can be combined.

Extensions. . .

- Efficient solvers for the semidefinite relaxation (exploiting low rank, randomization, etc.)
- Subset selection is a simple extension of sparse PCA (see Moghaddam, Weiss & Avidan (2006*a*) for example). The methods described here apply there too.
- Find better matrices with RI property.

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.

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', *Proceedings of the National Academy of Sciences* **102**(27), 9446–9451.

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.

Moghaddam, B., Weiss, Y. & Avidan, S. (2006*a*), Generalized spectral bounds for sparse LDA, *in* 'International Conference on Machine Learning'.

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

Sriperumbudur, B., Torres, D. & Lanckriet, G. (2007), Sparse eigen methods by d.c. programming, *in* 'ICML'.

Zou, H., Hastie, T. & Tibshirani, R. (2004), 'Sparse principal component analysis', *Journal of Computational and Graphical Statistics* .