High-dimensional data: Some challenges and recent progress

Martin Wainwright

UC Berkeley
Departments of Statistics, and EECS

Based on joint work with:

Alekh Agarwahl (UC Berkeley)
John Lafferty (Univ. Chicago)
Sahand Negahban (UC Berkeley)
Pradeep Ravikumar (UT Austin)

Era of massive data sets

- science and engineering in 21st century:
  - rapid technological advances (sensors, storage, computing etc.)
  - tremendous amounts of data being collected

Era of massive data sets

- science and engineering in 21st century:
  - rapid technological advances (sensors, storage, computing etc.)
  - tremendous amounts of data being collected
- many examples:
  - molecular biology: genomics, proteomics, etc.
  - neuroscience: fMRI, PET, EEG,, multi-electrode recording etc.
  - astronomy: Sloan digital sky survey, Large synoptic survey telescope etc.
  - consumer preference data: Netflix, Amazon, etc.
  - geosciences: hyperspectral imaging
  - financial data: stocks, bonds, currencies, derivatives etc.

Era of massive data sets

- science and engineering in 21st century:
  - rapid technological advances (sensors, storage, computing etc.)
  - tremendous amounts of data being collected
- many examples:
  - molecular biology: genomics, proteomics, etc.
  - neuroscience: fMRI, PET, EEG,, multi-electrode recording etc.
  - astronomy: Sloan digital sky survey, Large synoptic survey telescope etc.
  - consumer preference data: Netflix, Amazon, etc.
  - geosciences: hyperspectral imaging
  - financial data: stocks, bonds, currencies, derivatives etc.
- a wealth of data.....yet a paucity of information

Era of massive data sets

- science and engineering in 21st century:
  - rapid technological advances (sensors, storage, computing etc.)
  - tremendous amounts of data being collected
- many examples:
  - molecular biology: genomics, proteomics, etc.
  - neuroscience: fMRI, PET, EEG,, multi-electrode recording etc.
  - astronomy: Sloan digital sky survey, Large synoptic survey telescope etc.
  - consumer preference data: Netflix, Amazon, etc.
  - geosciences: hyperspectral imaging
  - financial data: stocks, bonds, currencies, derivatives etc.
- a wealth of data.....yet a paucity of information
- for statisticians: many exciting challenges and opportunities!

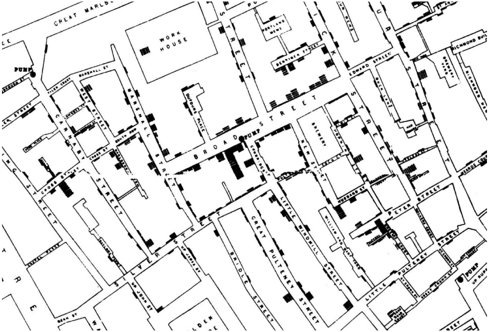
A story in three parts

1. Graphical models
   - Motivating applications: epidemiology, biology, social networks
   - Problem of model selection
   - Neighborhood-based discovery

2. Exploiting low-rank structure
   - Motivating applications: Recommender systems and collaborative filtering
   - Nuclear norm as a rank surrogate

3. Matrix decomposition problems
   - Motivating applications: robust PCA, security issues, hidden variables
   - Sparse plus low-rank: a simple relaxation

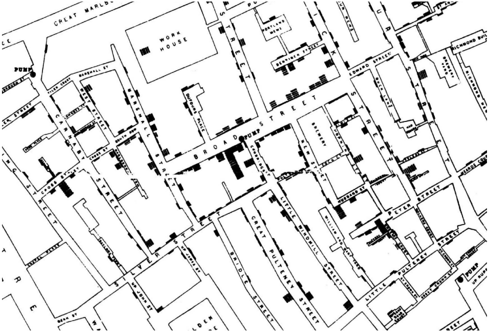
Epidemiological networks



(a) Cholera epidemic (London, 1854)
Snow, 1855

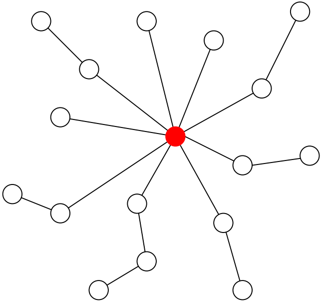
- network structure associated with spread of disease

Epidemiological networks



(a) Cholera epidemic (London, 1854)

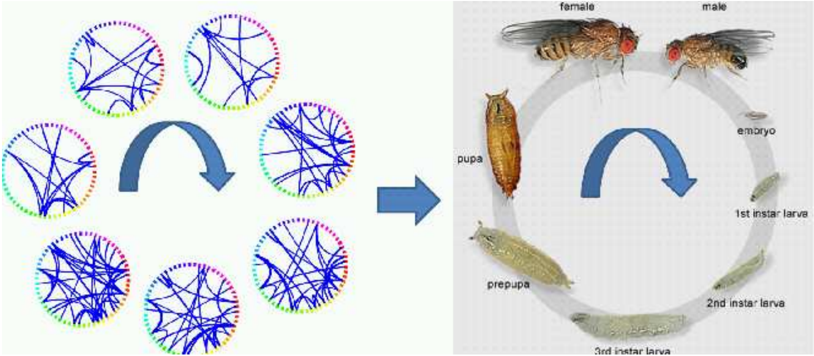
Snow, 1855



(b) "Spoke-hub" network

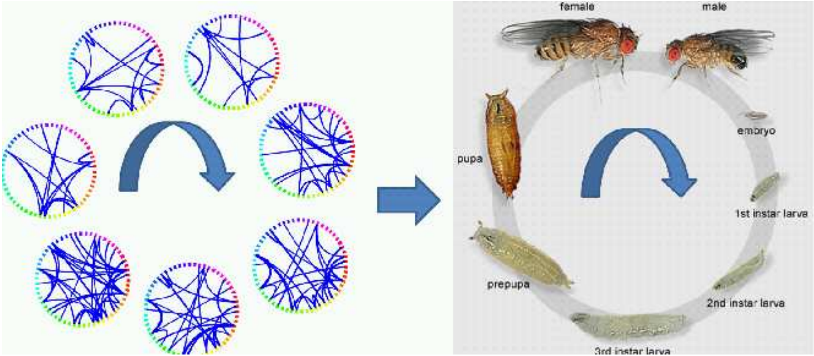
- network structure associated with spread of disease
- useful diagnostic information: contaminated water from Broad Street pump

Biological networks



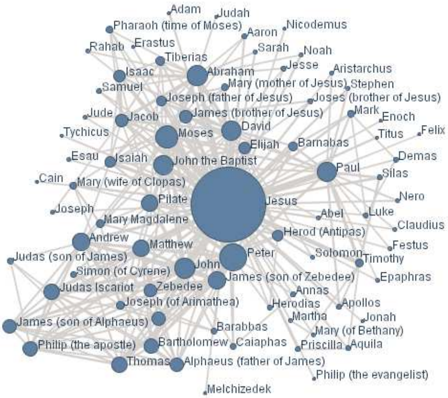
- gene networks during Drosophila life cycle (Ahmed & Xing, PNAS, 2009)

Biological networks



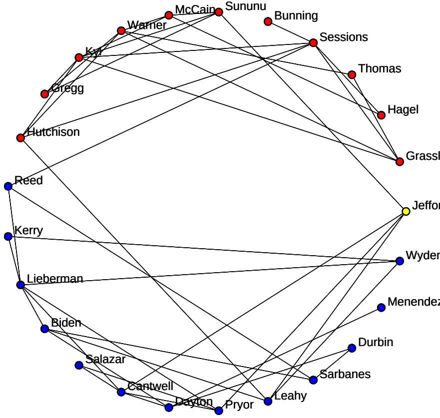
- gene networks during Drosophila life cycle (Ahmed & Xing, PNAS, 2009)
- many other examples:
  - protein networks
  - phylogenetic trees
  - neural networks for brain-machine interfaces (e.g., Carmena et al., 2009)

Social networks



(a) Biblical characters

www.esv.org



(b) US senators (2004-2006)

(Ravikumar, W. & Lafferty, 2006)

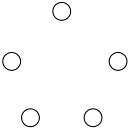
Example: Voting and graphical models

Vote of person s: $$x_s = \begin{cases} +1 & \text{if individual } s \text{ votes ``yes''} \\ -1 & \text{if individual } s \text{ votes ``no''} \end{cases}$$

Example: Voting and graphical models

Vote of person s: $$x_s = \begin{cases} +1 & \text{if individual } s \text{ votes "yes"} \\ -1 & \text{if individual } s \text{ votes "no"} \end{cases}$$

(1) Independent voting

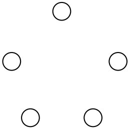


$$\mathbb{P}(x_1, \ldots, x_5) \propto \prod_{s=1}^{5} \exp(\theta_s x_s)$$

Example: Voting and graphical models

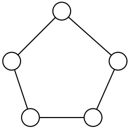
Vote of person s: $\quad x_s = \begin{cases} +1 & \text{if individual } s \text{ votes "yes"} \\ -1 & \text{if individual } s \text{ votes "no"} \end{cases}$

(1) Independent voting



$$\mathbb{P}(x_1, \ldots, x_5) \propto \prod_{s=1}^{5} \exp(\theta_s x_s)$$

(2) Cycle-based voting



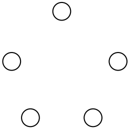
$$\mathbb{P}(x_1, \ldots, x_5) \propto \prod_{s=1}^{5} \exp(\theta_s x_s) \prod_{(s,t) \in C} \exp(\theta_{st} x_s \, x_t)$$

Example: Voting and graphical models

Vote of person s: $$x_s = \begin{cases} +1 & \text{if individual } s \text{ votes "yes"} \\ -1 & \text{if individual } s \text{ votes "no"} \end{cases}$$

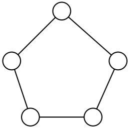
(1) Independent voting

$$\mathbb{P}(x_1, \ldots, x_5) \propto \prod_{s=1}^{5} \exp(\theta_s x_s)$$



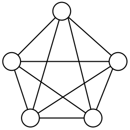
(2) Cycle-based voting

$$\mathbb{P}(x_1, \ldots, x_5) \propto \prod_{s=1}^{5} \exp(\theta_s x_s) \prod_{(s,t) \in C} \exp(\theta_{st} x_s \, x_t)$$

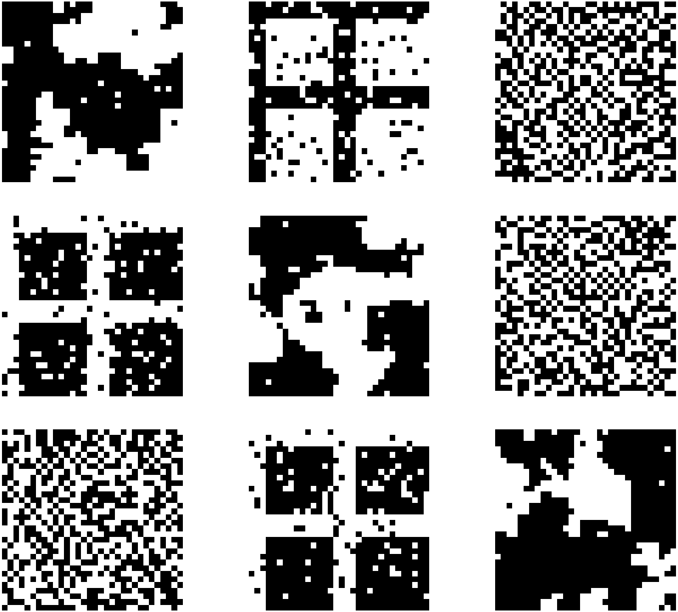


(3) Full clique voting

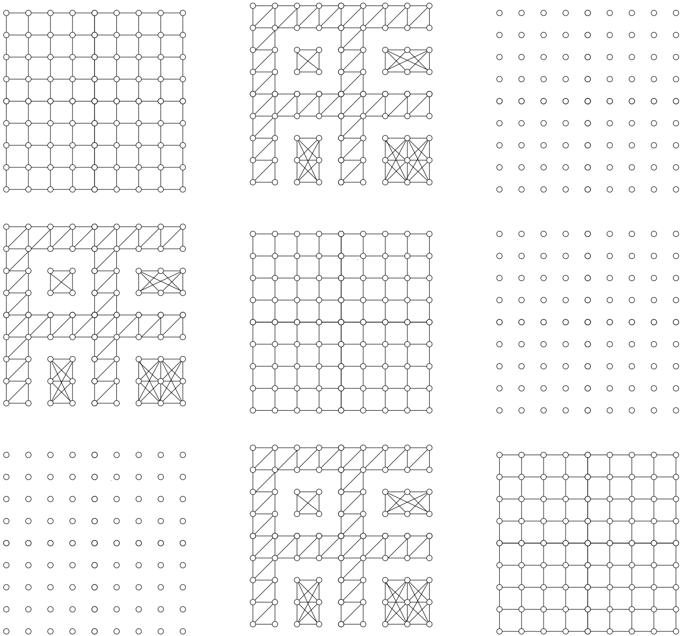
$$\mathbb{P}(x_1, \ldots, x_5) \propto \prod_{s=1}^{5} \exp(\theta_s x_s) \prod_{s \neq t} \exp(\theta_{st} x_s x_t)$$



Possible voting patterns



Underlying graphs

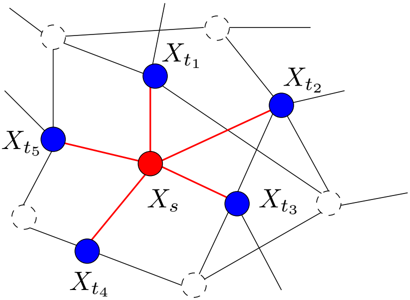


Markov property and neighborhood structure

- Markov properties encode neighborhood structure:

$$\underbrace{(X_s \quad | \quad X_{V\backslash s})}_{\text{Condition on full graph}} \stackrel{d}{=} \underbrace{(X_s \quad | \quad X_{N(s)})}_{\text{Condition on Markov blanket}}$$

$$N(s) = \{t_1, t_2, t_3, t_4, t_5\}$$



- basis of pseudolikelihood method (Besag, 1974)
- used for Gaussian model selection (Meinshausen & Buhlmann, 2006)

Graph selection via neighborhood regression

Ravikumar, Wainwright & Lafferty, 2006, 2010

Key: Graph recovery G equivalent to recovering neighborhood sets $N(s)$.

Method: Based on n samples:

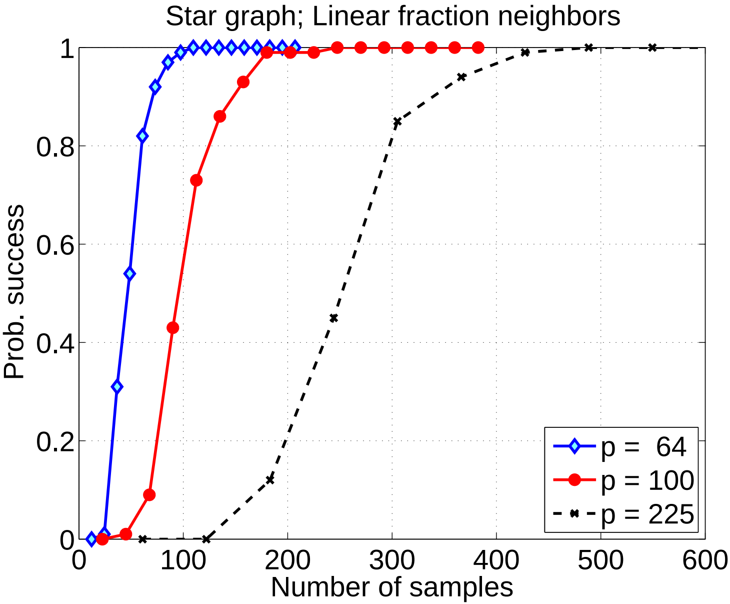
1. For each node s, predict X_s based on other variables $X_{\backslash s}$:

$$\widehat{\theta}[s] \quad := \quad \arg\min_{\theta \in \mathbb{R}^{p-1}} \left\{ -\frac{1}{n} \sum_{i=1}^{n} \underbrace{\log \mathbb{P}(\theta; X_{\backslash s}^{(i)})}_{\text{negative log likelihood}} \quad + \quad \lambda_{n\,n} \underbrace{\sum_{t \in V \backslash \{s\}} |\theta_{st}|}_{\ell_1 \text{ regularization}} \right\}$$

2. Estimate local neighborhood $\widehat{N}(s)$ by extracting non-zero positions within $\widehat{\theta}[s]$.

3. Combine the neighborhood estimates to form a graph estimate $\widehat{G}$.

Empirical behavior: Unrescaled plots



Empirical behavior: Appropriately rescaled

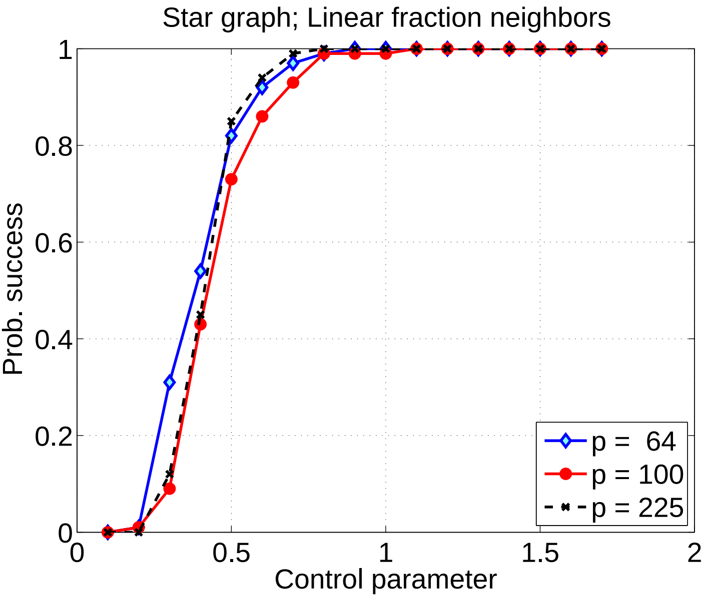


Illustration: Social network of US senators

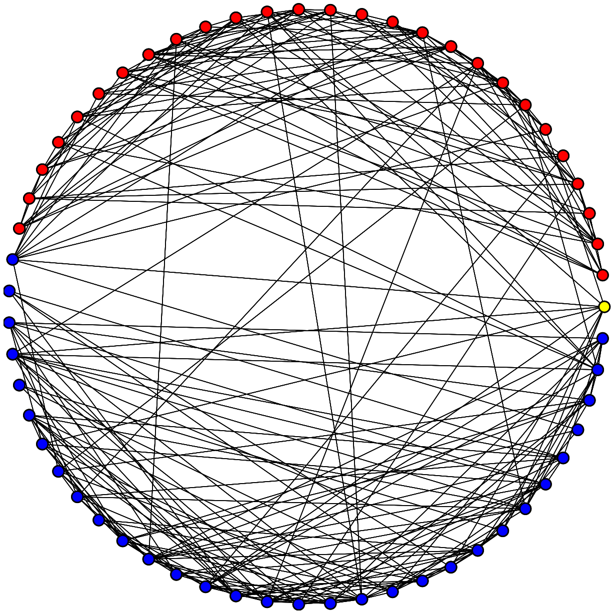
- originally studied by Bannerjee, Aspremont and El Ghaoui (2008)
- discrete data set of voting records for $p = 100$ senators:

$$X_{ij} = \begin{cases} +1 & \text{if senator } i \text{ voted yes on bill } j \\ -1 & \text{otherwise.} \end{cases}$$

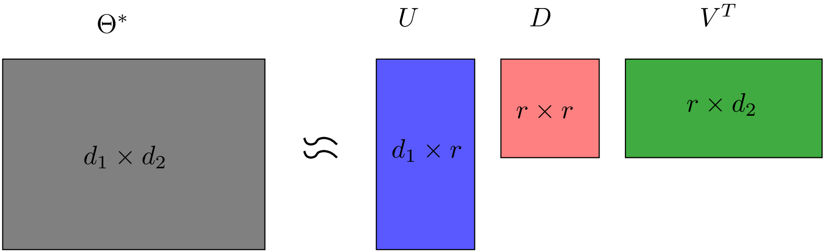
- full data matrix $X \in \mathbb{R}^{n \times p}$ with $n = 542$:

$$X = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ X_{31} & X_{32} & \cdots & X_{3p} \\ \vdots & \cdots & \cdots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{np} \end{bmatrix}$$

Estimated senator network (subgraph of 55)



§2. (Nearly) low-rank matrices

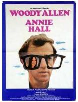
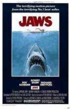
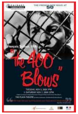
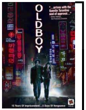
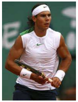


Matrix $\Theta^* \in \mathbb{R}^{d_1 \times d_2}$ with rank $r \ll \min\{d_1, d_2\}$.

Singular value decomposition:

- matrix of left singular vectors $U \in \mathbb{R}^{d_1 \times r}$
- matrix of right singular vectors $V \in \mathbb{R}^{d_2 \times r}$
- singular values $\sigma_1(\Theta^*) \geq \sigma_2(\Theta^*) \geq \cdots \geq \sigma_r(\Theta^*) \geq 0$.

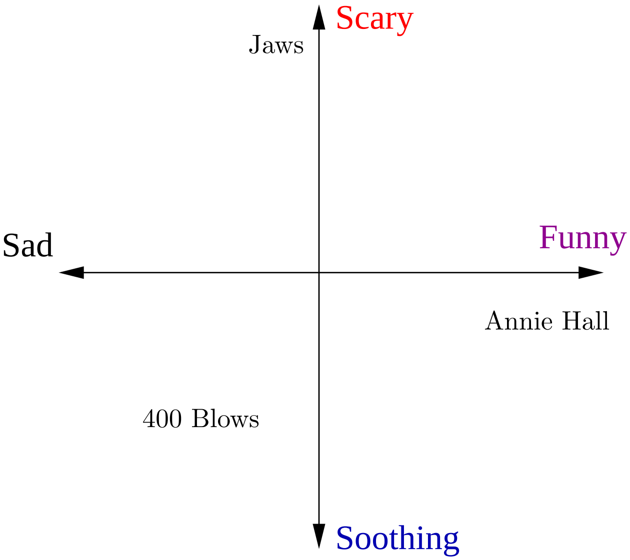
Example: Matrix completion

				$\ldots$	$\ldots$	
	4	$*$	3	$\ldots$	$\ldots$	$*$
	3	5	$*$	$\ldots$	$\ldots$	2
	5	4	3	$\ldots$	$\ldots$	3
	2	$*$	$*$	$\ldots$	$\ldots$	1

Universe of d_1 individuals and d_2 films   Observe $n \ll d_1 d_2$ ratings

Typical numbers for Netflix: $d_1 \approx 10^5$–10^8 and $d_2 \approx 10^6$–10^{10}

Geometry of low-rank model



Nuclear norm as a rank surrogate

- Rank as an ℓ_0-"norm" on vector of singular values:

$$\text{rank}(\Theta^*) = \sum_{j=1}^{d} \mathbb{I}\big[\sigma_j(\Theta) \neq 0\big] \qquad \text{where } d = \min\{d_1, d_2\}.$$

- Non-convexity: rank constraints computationally hard.

Nuclear norm as a rank surrogate

- Rank as an ℓ_0-"norm" on vector of singular values:

$$\text{rank}(\Theta^*) = \sum_{j=1}^{d} \mathbb{I}\big[\sigma_j(\Theta) \neq 0\big] \qquad \text{where } d = \min\{d_1, d_2\}.$$

- Non-convexity: rank constraints computationally hard.

- Nuclear norm: convex relaxation of rank:

$$\|\Theta\|_{\text{nuc}} = \sum_{j=1}^{d} \sigma_j(\Theta).$$

Nuclear norm as a rank surrogate

- Rank as an ℓ_0-"norm" on vector of singular values:

$$\text{rank}(\Theta^*) = \sum_{j=1}^{d} \mathbb{I}\big[\sigma_j(\Theta) \neq 0\big] \qquad \text{where } d = \min\{d_1, d_2\}.$$

- Non-convexity: rank constraints computationally hard.

- Nuclear norm: convex relaxation of rank:

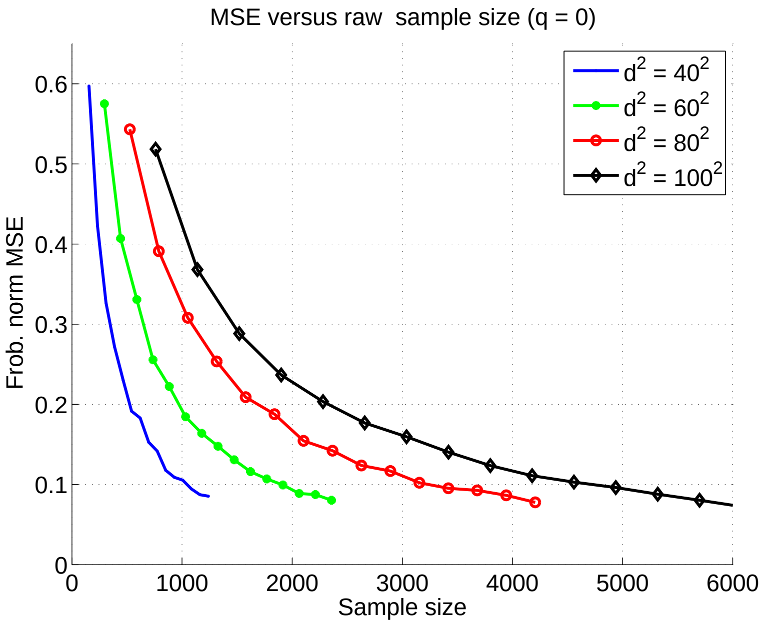
$$|||\Theta|||_{\text{nuc}} = \sum_{j=1}^{d} \sigma_j(\Theta).$$

- Estimator for matrix completion:

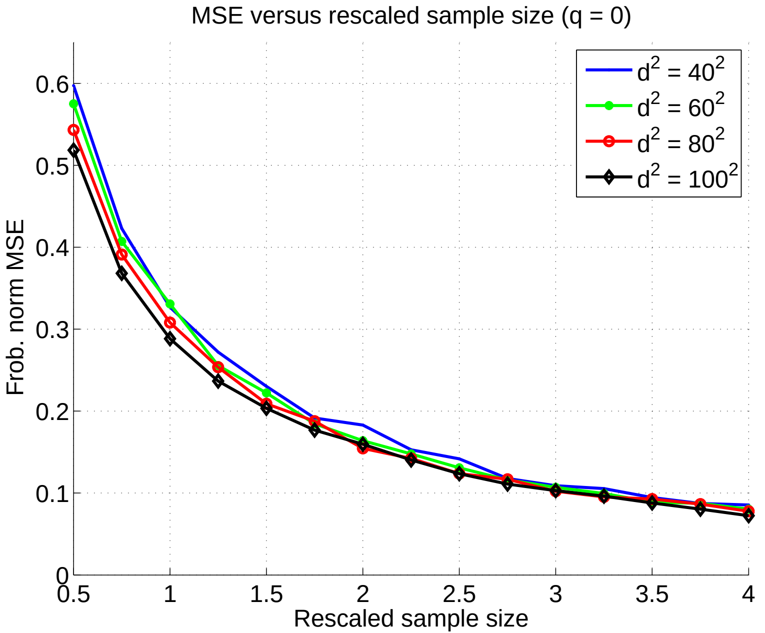
$$\widehat{\Theta} \in \arg \min_{\Theta \in \mathbb{R}^{d_1 \times d_2}} \Big\{ \sum_{(a,b) \in \Omega} \big(Y_{ab} - \Theta_{ab}\big)^2 + \lambda_n |||\Theta|||_{\text{nuc}} \Big\}$$

(Fazel, 2001; Srebro et al., 2004; Candes & Recht, 2009; Negahban & Wainwright, 2010)

Noisy matrix completion (unrescaled)



Noisy matrix completion (rescaled)



A simple iterative algorithm

Projected gradient descent over nuclear norm ball with stepsize $\alpha > 0$:

1. Compute gradient at current iterate Θ^t

$$[\nabla \mathcal{L}(\Theta^t)]_{ab} = \begin{cases} \Theta^t_{ab} - Y_{ab} & \text{if entry } (a,b) \text{ observed.} \\ 0 & \text{otherwise.} \end{cases}$$

A simple iterative algorithm

Projected gradient descent over nuclear norm ball with stepsize $\alpha > 0$:

1. Compute gradient at current iterate Θ^t

$$[\nabla \mathcal{L}(\Theta^t)]_{ab} = \begin{cases} \Theta^t_{ab} - Y_{ab} & \text{if entry } (a,b) \text{ observed.} \\ 0 & \text{otherwise.} \end{cases}$$

2. Compute singular value decomposition of matrix $\Gamma = \Theta^t - \alpha \nabla \mathcal{L}(\Theta^t)$.

A simple iterative algorithm

Projected gradient descent over nuclear norm ball with stepsize $\alpha > 0$:

1. Compute gradient at current iterate Θ^t

$$[\nabla \mathcal{L}(\Theta^t)]_{ab} = \begin{cases} \Theta^t_{ab} - Y_{ab} & \text{if entry } (a,b) \text{ observed.} \\ 0 & \text{otherwise.} \end{cases}$$

2. Compute singular value decomposition of matrix $\Gamma = \Theta^t - \alpha \nabla \mathcal{L}(\Theta^t)$.

3. Return Θ^{t+1} by soft-thresholding the singular values of Γ at level λ_n.

Implemented by Mazumber, Hastie & Tibshirani, 2009

A simple iterative algorithm

Projected gradient descent over nuclear norm ball with stepsize $\alpha > 0$:

1. Compute gradient at current iterate Θ^t

$$[\nabla \mathcal{L}(\Theta^t)]_{ab} = \begin{cases} \Theta^t_{ab} - Y_{ab} & \text{if entry } (a,b) \text{ observed.} \\ 0 & \text{otherwise.} \end{cases}$$

2. Compute singular value decomposition of matrix $\Gamma = \Theta^t - \alpha \nabla \mathcal{L}(\Theta^t)$.

3. Return Θ^{t+1} by soft-thresholding the singular values of Γ at level λ_n.

Implemented by Mazumber, Hastie & Tibshirani, 2009

Question:

How quickly does this algorithm converge?

A simple iterative algorithm

Projected gradient descent over nuclear norm ball with stepsize $\alpha > 0$:

1. Compute gradient at current iterate Θ^t

$$[\nabla \mathcal{L}(\Theta^t)]_{ab} = \begin{cases} \Theta^t_{ab} - Y_{ab} & \text{if entry } (a,b) \text{ observed.} \\ 0 & \text{otherwise.} \end{cases}$$

2. Compute singular value decomposition of matrix $\Gamma = \Theta^t - \alpha \nabla \mathcal{L}(\Theta^t)$.
3. Return Θ^{t+1} by soft-thresholding the singular values of Γ at level λ_n.

Implemented by Mazumber, Hastie & Tibshirani, 2009

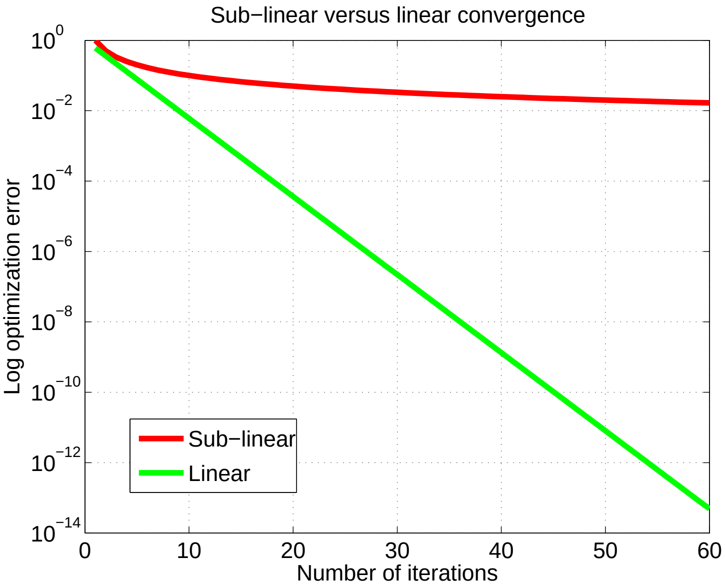
Question:

How quickly does this algorithm converge?

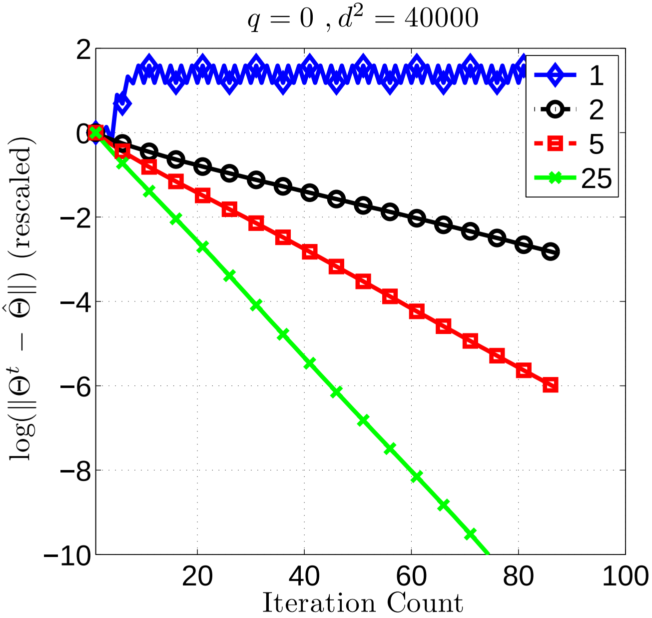
Without additional structure, would expect slow (sub-linear) convergence:

$$|||\Theta^t - \widehat{\Theta}|||_F^2 \approx \frac{1}{t}.$$

Sub-linear versus linear convergence

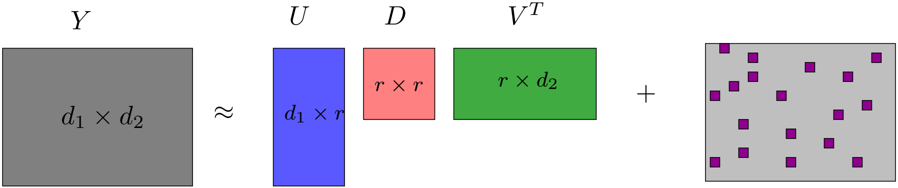


Fast convergence rates for matrix completion



§3. Matrix decomposition: Low-rank plus sparse

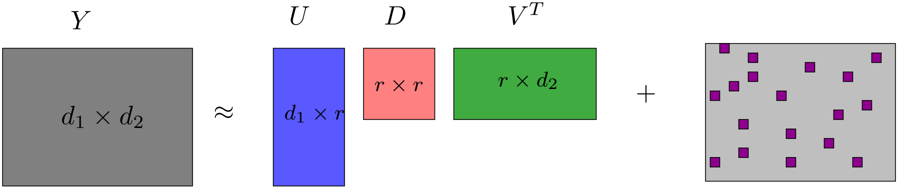
Matrix Y can be (approximately) decomposed into sum:



$$Y = \underbrace{\Theta^*}_{\text{Low-rank component}} + \underbrace{\Gamma^*}_{\text{Sparse component}}$$

§3. Matrix decomposition: Low-rank plus sparse

Matrix Y can be (approximately) decomposed into sum:

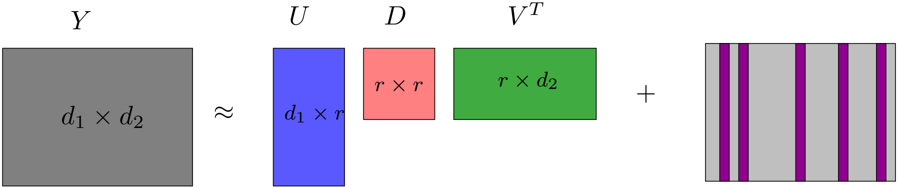


$$Y = \underbrace{\Theta^*}_{\text{Low-rank component}} + \underbrace{\Gamma^*}_{\text{Sparse component}}$$

- exact decomposition: initially studied by Chandrasekaran et al., 2009
- Various applications:
  - robust collaborative filtering
  - graphical model selection with hidden variables
  - image/video segmentation

Matrix decomposition: Low-rank plus column sparse

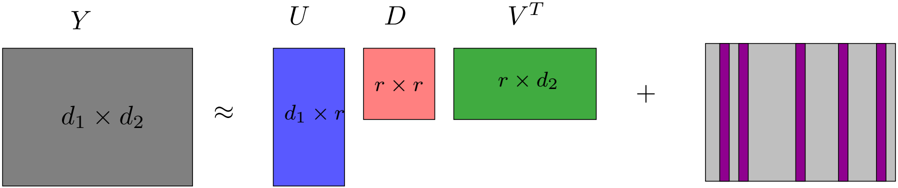
Matrix Y can be (approximately) decomposed into sum:



$$Y = \underbrace{\Theta^*}_{\text{Low-rank component}} + \underbrace{\Gamma^*}_{\text{Column sparse component}}$$

Matrix decomposition: Low-rank plus column sparse

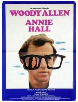
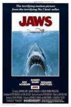
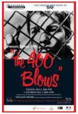
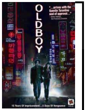
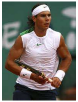
Matrix Y can be (approximately) decomposed into sum:



$$Y = \underbrace{\color{blue}{\Theta^*}}_{\text{Low-rank component}} + \underbrace{\color{purple}{\Gamma^*}}_{\text{Column sparse component}}$$

- exact decomposition: initially studied by Xu et al., 2010
- Various applications:
  - robust collaborative filtering
  - robust principal components analysis

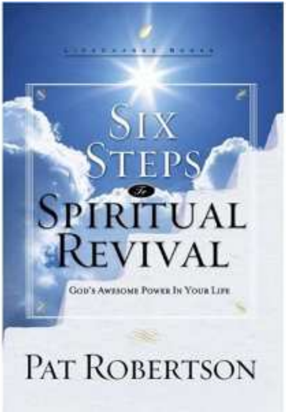
Example: Collaborative filtering

				$\dots$	$\dots$	
	4	$*$	3	$\dots$	$\dots$	$*$
	3	5	$*$	$\dots$	$\dots$	2
	5	4	3	$\dots$	$\dots$	3
	2	$*$	$*$	$\dots$	$\dots$	1

Universe of d_1 individuals and d_2 films   Observe $n \ll d_2 d_2$ ratings

(e.g., Srebro, Alon & Jaakkola, 2004)

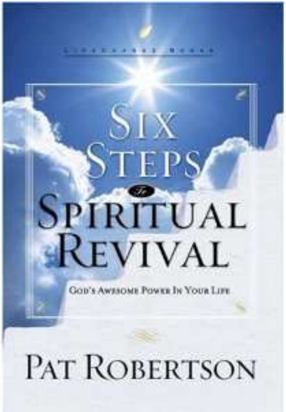
Security and robustness issues



Spiritual guide

Break-down of Amazon recommendation system (New York Times, 2002).

Security and robustness issues

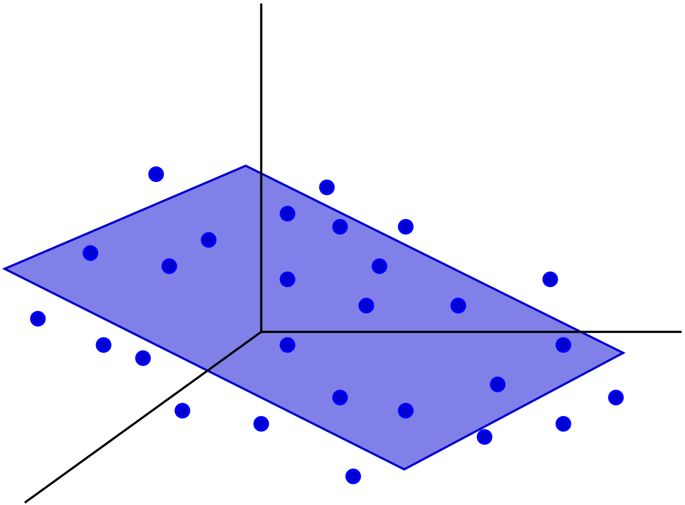


Spiritual guide

Sex manual

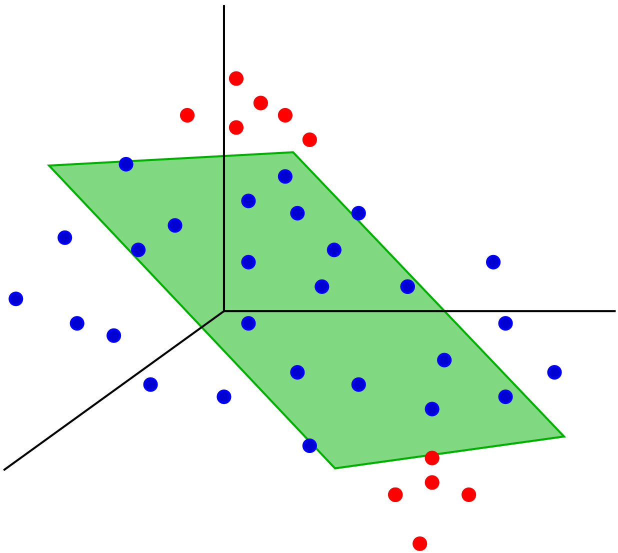
Break-down of Amazon recommendation system (New York Times, 2002).

Example: Robustness in PCA



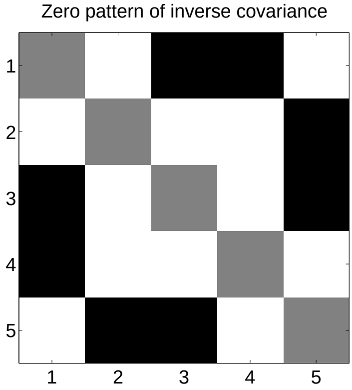
Standard PCA fits a low-rank matrix to a data matrix.

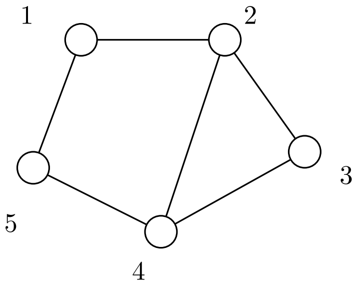
Example: Robustness in PCA



A small amount of data corruption can have a large influence.

Example: Structure of Gauss-Markov random fields

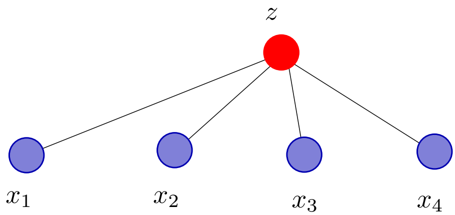




Multivariate Gaussian with graph-structured inverse covariance Γ^*:

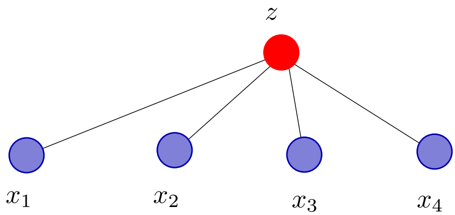
$$\mathbb{P}(x_1, x_2, \ldots, x_p) \propto \exp\big(-\frac{1}{2} x^T \Gamma^* x\big).$$

Gauss-Markov models with hidden variables



Problems with hidden variables: conditioned on hidden z, vector $x = (x_1, x_2, x_3, x_4)$ is Gauss-Markov.

Gauss-Markov models with hidden variables



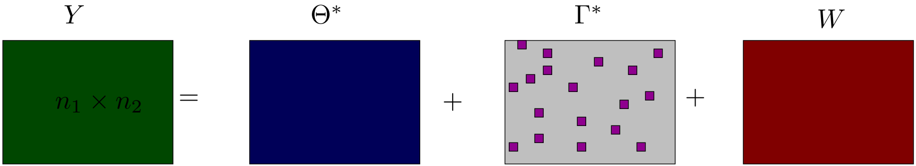
Problems with hidden variables: conditioned on hidden z, vector $x = (x_1, x_2, x_3, x_4)$ is Gauss-Markov.

Inverse covariance of x satisfies {sparse, low-rank} decomposition:

$$\begin{bmatrix} 1-\mu & \mu & \mu & \mu \\ \mu & 1-\mu & \mu & \mu \\ \mu & \mu & 1-\mu & \mu \\ \mu & \mu & \mu & 1-\mu \end{bmatrix} = I_{4\times 4} - \mu \mathbf{1}\mathbf{1}^T.$$

(Chandrasekaran, Parrilo & Willsky, 2010)

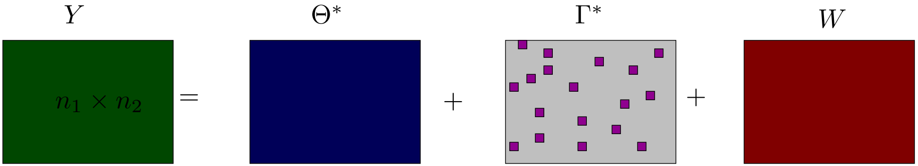
Method for noisy matrix decomposition



Given noisy observations:

$$Y = \Theta^* + \Gamma^* + W$$

Method for noisy matrix decomposition



Given noisy observations:

$$Y = \Theta^* + \Gamma^* + W$$

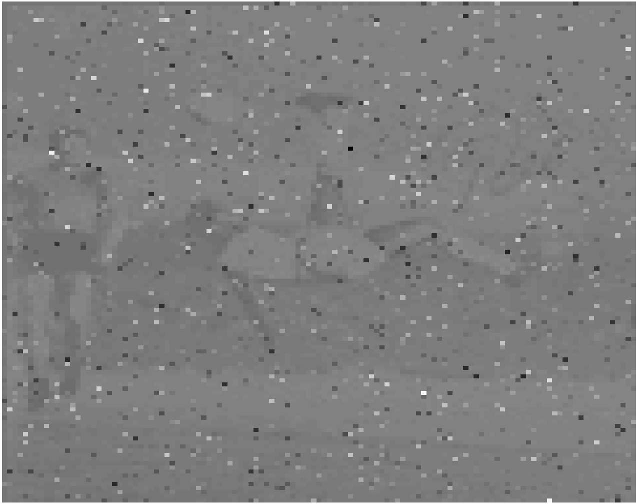
Solve convex program

$$(\widehat{\Theta}, \widehat{\Gamma}) \in \arg \min_{(\Theta, \Gamma)} \Big\{ |\!|\!| Y - (\Theta + \Gamma) |\!|\!|_{\text{frob}}^2 + \lambda_d |\!|\!| \Theta |\!|\!|_{\text{nuc}} + \mu_d \|\Gamma\|_1 \Big\}$$

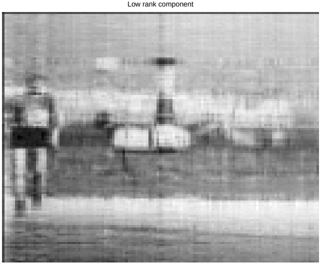
plus "spikiness" constraint $\|\Theta\|_\infty \leq \frac{\alpha_d}{\sqrt{d_1 d_2}}$.

Illustration

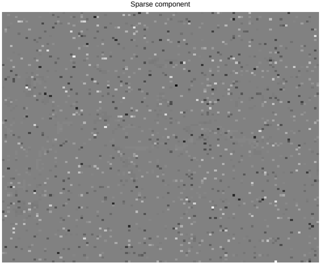
Original observations



Illustration



Low rank component



Sparse component

Noise matrix W

Summary

- characteristics of modern data sets:
  - large-scale: many samples, many predictors
  - high-dimensional: data dimension may exceed sample size
- challenges and opportunities for statisticians:
  - how to model low-dimensional structure?
  - new theory: non-asymptotic, allowing for high-dimensional scaling
  - closer coupling between statistical and computational concerns

Summary

- characteristics of modern data sets:
  - large-scale: many samples, many predictors
  - high-dimensional: data dimension may exceed sample size
- challenges and opportunities for statisticians:
  - how to model low-dimensional structure?
  - new theory: non-asymptotic, allowing for high-dimensional scaling
  - closer coupling between statistical and computational concerns

---

Some references:

- High-dimensional Ising model selection using ℓ_1-regularized logistic regression (2010). Annals of Statistics, 38(3): 1287–1317. With P. Ravikumar and J. Lafferty.
- Estimation rates of (near) low-rank matrices with noise and high-dimensional scaling (2011). Annals of Statistics, 39(2): 1069–1097. With S. Negahban.
- Restricted strong convexity and (weighted) matrix completion: Optimal bounds with noise. arxiv.org/abs/0112.5100, September 2010, With S. Negahban.
- Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. http://arxiv.org/abs/1102.4807, February 2011. With A. Agarwal and S. Negahban.