Methods for covariance and inverse covariance estimation in high dimension

Ecole ETICS 2016

Stéphane Gaïffas

1. Covariance estimation
   - Estimation of covariance matrices
   - Random matrices with independent entries
   - Random matrices with independent rows
   - Bernstein's inequality for random matrices

2. Inverse covariance / Graphical Gaussian Model
   - A glimpse of graphical modelling / graph theory
   - Graphical Gaussian Model
   - Sparse estimation

3. Some tools from convex optimization
   - Some tools
   - Proximal gradient descent

1. Covariance estimation
   - Estimation of covariance matrices
   - Random matrices with independent entries
   - Random matrices with independent rows
   - Bernstein's inequality for random matrices

2. Inverse covariance / Graphical Gaussian Model
   - A glimpse of graphical modelling / graph theory
   - Graphical Gaussian Model
   - Sparse estimation

3. Some tools from convex optimization
   - Some tools
   - Proximal gradient descent

Problem

- Let a random vector $X = (X^1, \ldots, X^d) \in \mathbb{R}^d$
- The covariance matrix is

$$\Sigma = \mathbb{E}[(X - \mathbb{E}X)(X - \mathbb{E}X)^\top]$$

- How to estimate Σ?

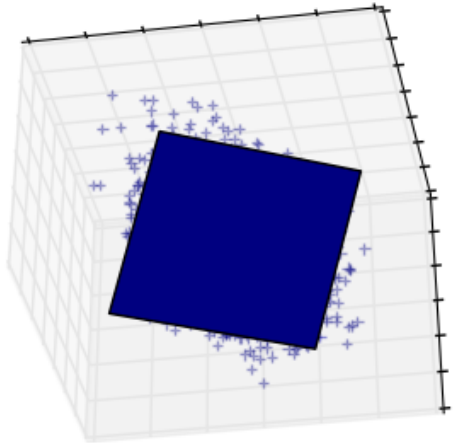
Given n i.i.d copies $X_1, \ldots, X_n$ of X, the canonical estimator is the empirical covariance

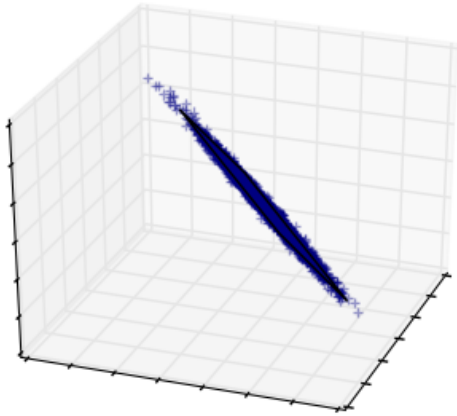
$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})(X_i - \bar{X})^\top \tag{1}$$

where $\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$

Applications

- Are countess!
- Principal Component Analysis (PCA)
- Linear / Quadratic discriminant analysis (LDA/QDA)
- Independance and conditional independence (graphical models, more on that later...)





Estimation of the covariance matrix

For same of simplicity, assume $\mathbb{E}X = 0$ (centering with $\bar{X}$)

- $\Sigma = \mathbb{E}XX^\top = (\mathrm{cov}(X^j, X^k))_{1 \le j,k \le d}$
- Σ is a symmetric, positive semi-definite (SDP) $d \times d$ matrix
- If $\Sigma = I$ we say that X is isotropic. Any random vector X can be made isotropic using the linear transformation $\Sigma^{-1/2}X$
- $\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^n X_i X_i^\top$
- Hopefully $\hat{\Sigma}$ approximates Σ well
- $\hat{\Sigma}$ is a *random matrix*

Covariance estimation problem

What is the minimum sample size $n = n(p)$ that guarantees that with high probability, $\hat{\Sigma}$ is close to Σ with a fixed accuracy in operator norm:

$$\|\hat{\Sigma} - \Sigma\|_{\mathrm{op}} \le \varepsilon \|\Sigma\|_{\mathrm{op}}$$

Estimation of the covariance matrix

- Estimation problem related to the *spectrum* of random matrices (spectrum = set of singular values)
- Assume for simplicity that $\Sigma = I$
- Define the $n \times d$ matrix $\boldsymbol{X}$ obtained by stacking vectically $X_1, \ldots, X_n$ (independent rows)
- We can write

$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} X_i X_i^\top = \frac{1}{n} \boldsymbol{X}^\top \boldsymbol{X}$$

Estimation of the covariance matrix

- Now the desired property is

$$\|\hat{\Sigma} - I\|_{\mathrm{op}} \leq \varepsilon$$

is equivalent to saying that there is an almost isometric embedding $\mathbb{R}^d \to \mathbb{R}^n$:

$$(1-\varepsilon)\sqrt{n} \leq \|\boldsymbol{X}v\|_2 \leq (1+\varepsilon)\sqrt{n} \quad \text{for any } v \in S^{d-1}$$

- Equivalently, the singular values $\sigma_j(\boldsymbol{X}) = \sqrt{\lambda_j(\boldsymbol{X}^\top \boldsymbol{X})}$ are all close, and close to $\sqrt{n}$:

$$(1-\varepsilon)\sqrt{n} \leq \sigma_{\min}(\boldsymbol{X}) \leq \sigma_{\max}(\boldsymbol{X}) \leq (1+\varepsilon)\sqrt{n}$$

Question

- What are the random matrices with independent rows that are almost isometric embeddings?

Random matrices with independent entries

Random Matrix Theory (RMT) studies the asymptotic regime $n, d \to +\infty$

Theorem [Marchenko–Pastur 1967]
If $\boldsymbol{X}$ with i.i.d standard entries (mean zero, variance 1) and if $n, d \to +\infty$ and $n/d \to y \in [0, 1]$ one as

$$\frac{1}{d} \sum_{j=1}^{d} \mathbf{1}_{\frac{1}{n}\lambda_j(\boldsymbol{X}^\top \boldsymbol{X}) \leq x} \to F(x)$$

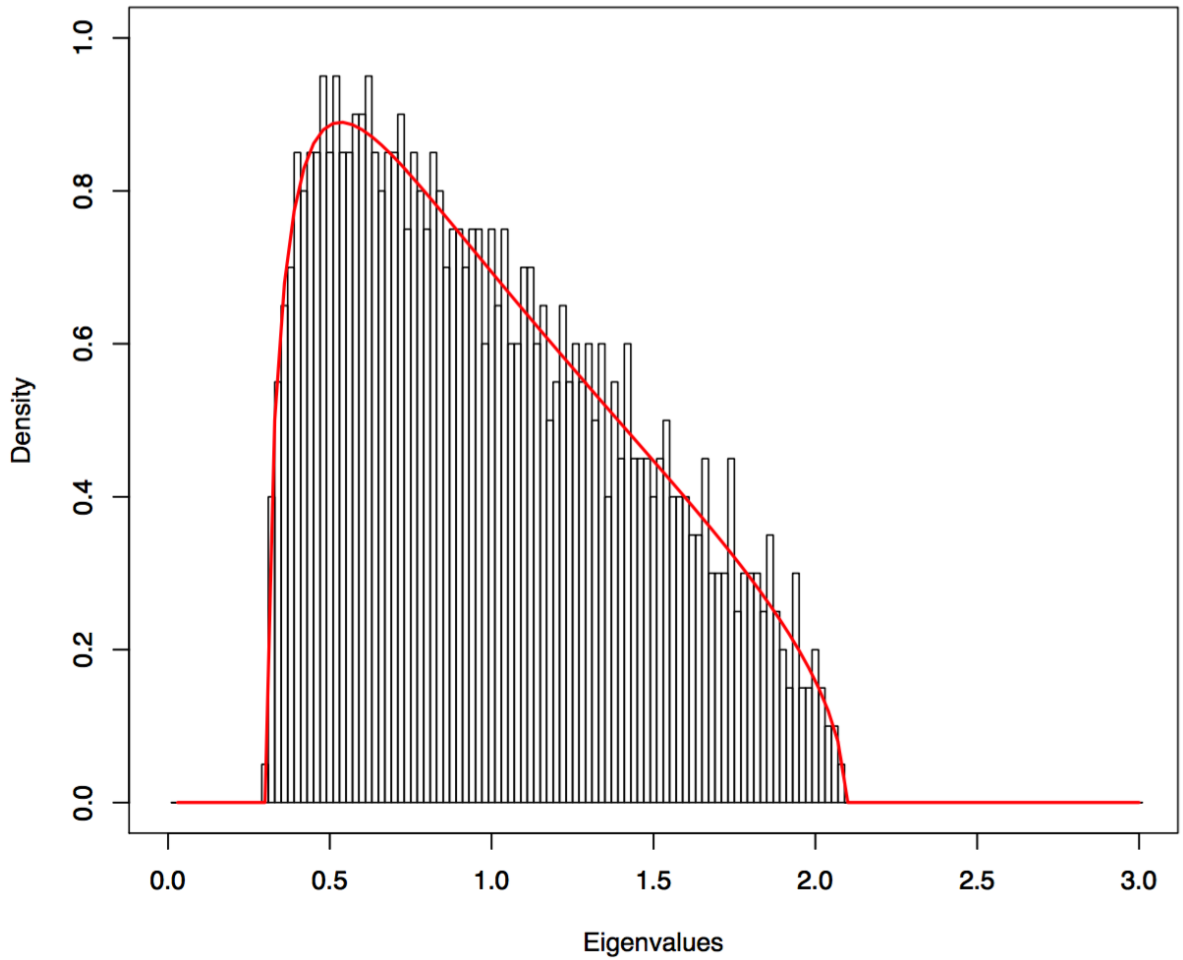
where

$$F'(x) = \frac{1}{2\pi x y} \sqrt{(b-x)(a-x)} \mathbf{1}_{[a,b]}(x)$$

with

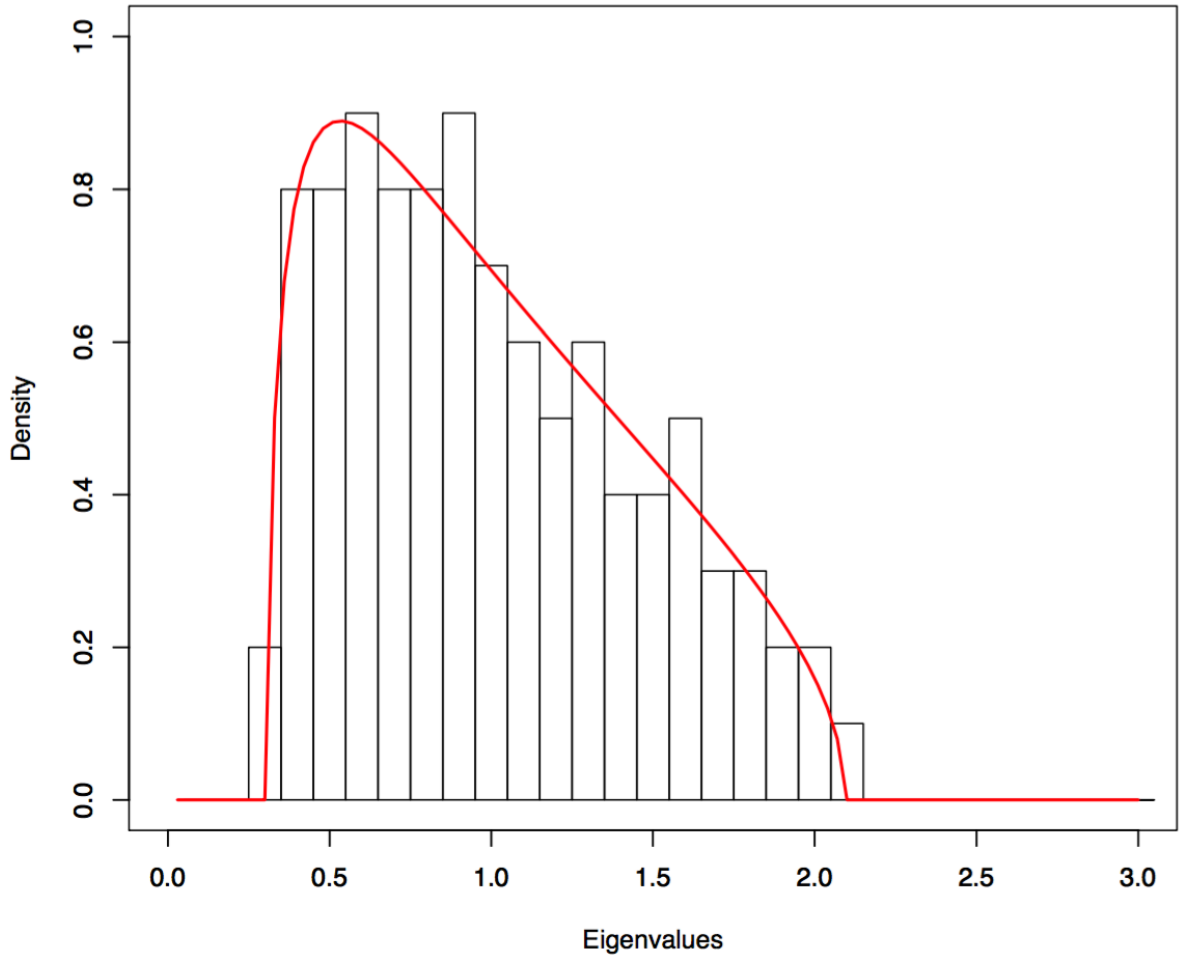
$$a = (1 - \sqrt{y})^2 \quad \text{and} \quad b = (1 + \sqrt{y})^2$$

Random matrices with independent entries



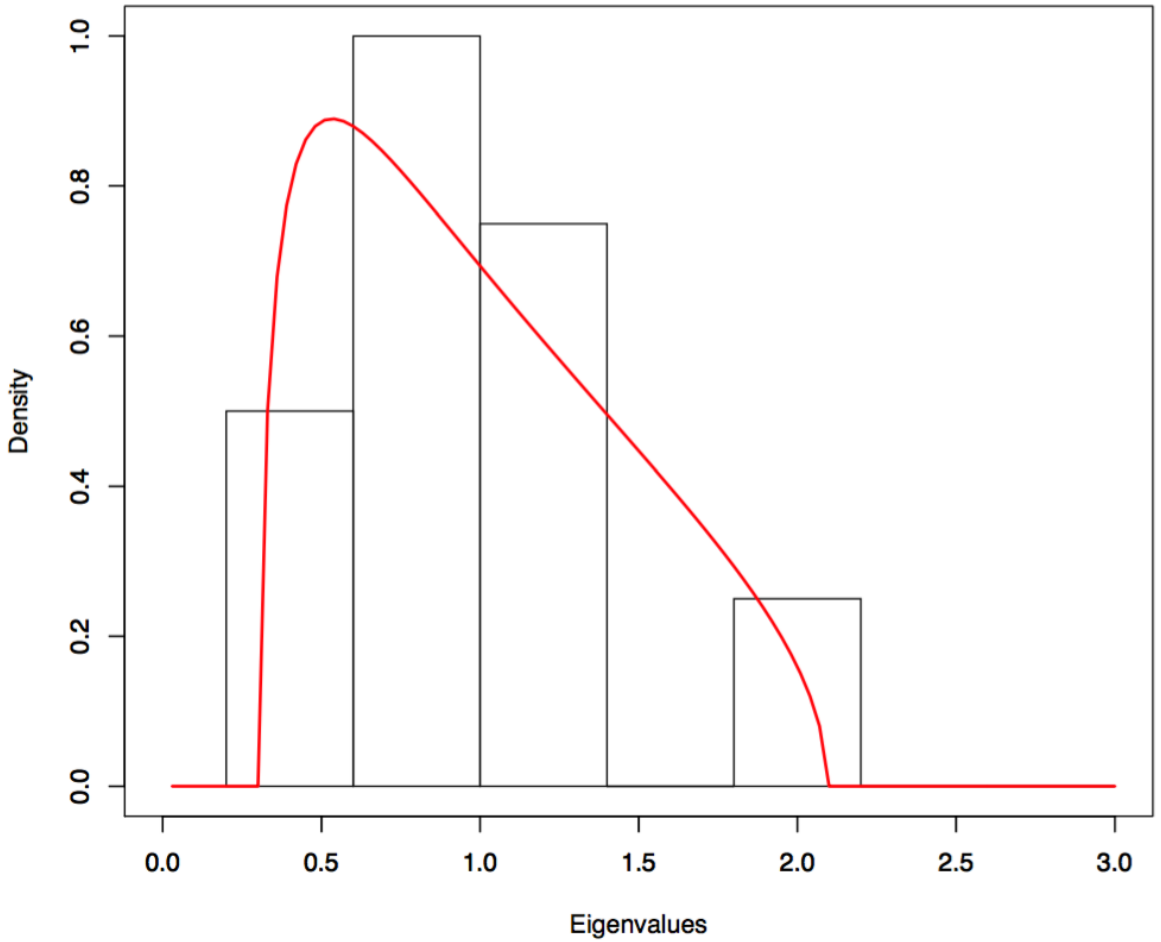
Marchenko–Pastur distribution $n = 1000, d = 5000$

Random matrices with independent entries



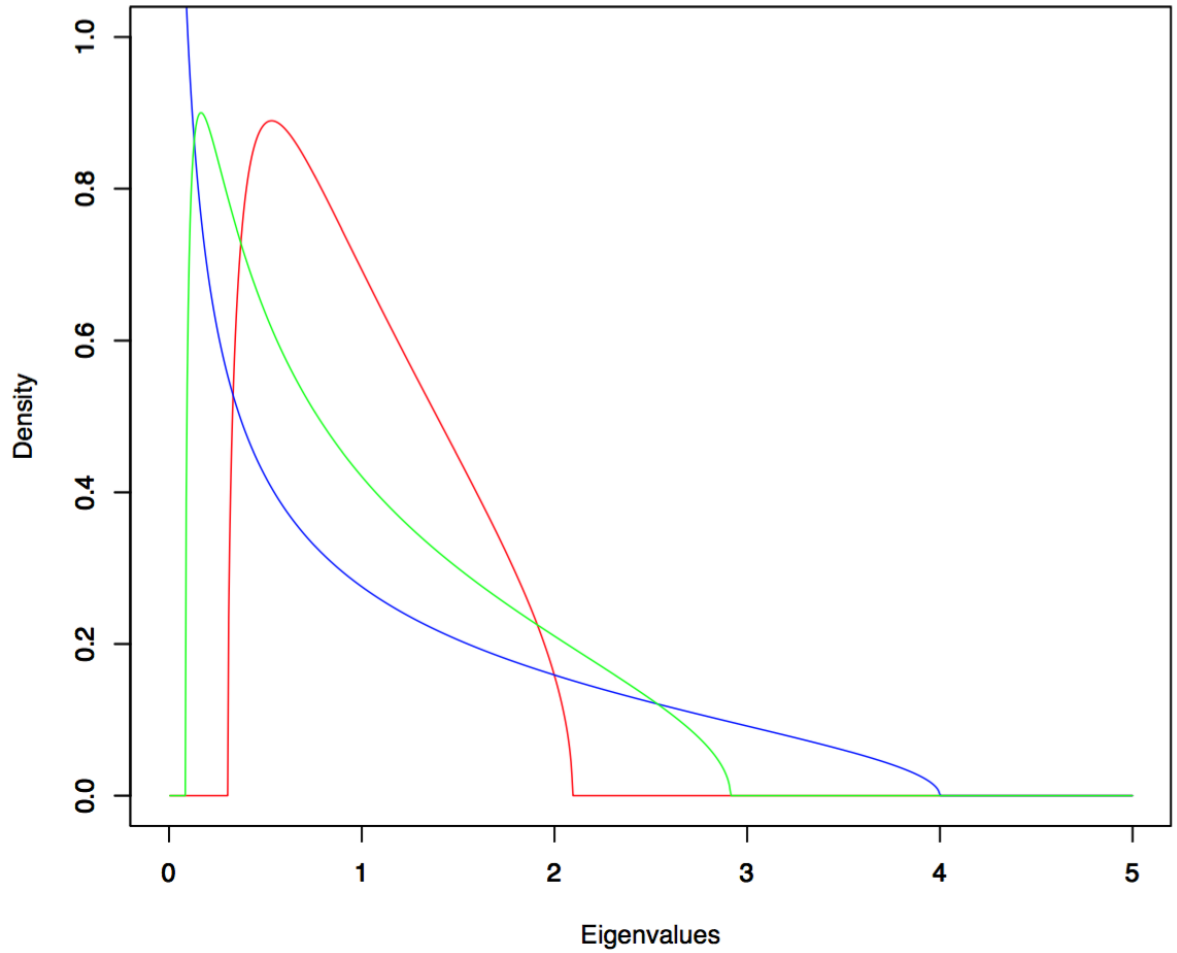
Marchenko–Pastur distribution $n = 100, d = 500$

Random matrices with independent entries



Marchenko–Pastur distribution $n = 10, d = 50$

Random matrices with independent entries



Marchenko–Pastur distribution $y = 1$ (blue), $y = 1/2$ (green), $y = 1/5$ (red)

Random matrices with independent entries

- Compact support for the singular values! No outliers
- Roughly speaking

$$\sigma_j\Big(\frac{1}{\sqrt{n}}\boldsymbol{X}\Big) \in \Big[1-\sqrt{\frac{d}{n}}, 1+\sqrt{\frac{d}{n}}\Big]$$

- Making n larger than d force both extreme values to be *close*: makes $\boldsymbol{X}$ an almost isometric embedding
- For matrices with i.i.d entries $n(d) \sim d$ is enough to estimate the covariance
- Taller is better: improves conditioning $\sigma_{max}(\boldsymbol{X})/\sigma_{min}(\boldsymbol{X})$
- **But:** most distributions do not have independent coordinates
- Only the rows of $\boldsymbol{X}$ are independent (samples)
- Non-asymptotic results? What assumptions?

Subgaussian distribution

A random vector $X \in \mathbb{R}^d$ is *subgaussian* if all marginals are subgaussian random variables:

$$\mathbb{P}[|X^\top v| \geq t] \leq 2e^{-ct^2} \quad \text{for any } v \in S^{d-1}$$

Similar definition for subexponential (replace t^2 by t)

Examples

- Standard Gaussian distribution is subgaussian
- Uniform distribution on a ball or a cube of unit volume is subgaussian
- Uniform distribution on any convex body (of unit volume) is sub-exponential (Brunn-Minkowski inequality, Borell's lemma)

Random matrices with independent rows

Theorem [Vershynin 2010]
Let $\boldsymbol{X}$ be $n \times d$ matrix with rows X_i independent subgaussian isotropic in $\mathbb{R}^d$. Then with large probability

$$\sqrt{n} - c\sqrt{d} \leq \sigma_{\min}(\boldsymbol{X}) \leq \sigma_{\max}(\boldsymbol{X}) \leq \sqrt{n} + c\sqrt{d}$$

- Entails that $\hat{\Sigma} = \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{X}$ approximates the actual covariance matrix I:

$$\|\hat{\Sigma} - I\|_{\text{op}} \leq c\sqrt{\frac{d}{n}}$$

- Positive answer to the question for independent subgaussian rows: sample size $n(d) \sim d$ is enough to estimate the covariance matrix using $\hat{\Sigma}$

Proof [$\varepsilon-$net argument]

- We show that $\|\boldsymbol{X}v\|_2^2$ is close to its expected value n for any $v \in S^{d-1}$
- But

$$\|\boldsymbol{X}v\|_2^2 = \sum_{i=1}^{n} (X_i^\top v)^2$$

  is a sum of *independent subexponential* random variables
- Use exponential deviation inequalities (Bernstein's inequality), to prove that $\|\boldsymbol{X}v\|_2^2 \approx n$ for a fixed $v \in S^{d-1}$
- Cover S^{d-1} by an ε-net and use a union bound

But

- Fails for moments weaker than subgaussian
- Different ideas for heavier tails
- Boundedness assumption: distribution of X is supported in a ball of radius $O(\sqrt{d})$. Note that for isotropic distribution $\mathbb{E}\|X\|_2^2 = d$

Random matrices with independent rows

Under no moment assumption, we can prove

Theorem [Bourgain 1999, Rudelson 2000]
Let $\boldsymbol{X}$ be $n \times d$ matrix with rows X_i independent isotropic in $\mathbb{R}^d$.
Then with large probability

$$\sqrt{n} - c\sqrt{d \log d} \leq \sigma_{\min}(\boldsymbol{X}) \leq \sigma_{\max}(\boldsymbol{X}) \leq \sqrt{n} + c\sqrt{d \log d}$$

- $\log d$ is needed (uniform distribution on d orthogonal vectors)
- Gives again a control of the form

$$\|\hat{\Sigma} - I\|_{\text{op}} \leq c\sqrt{\frac{d \log d}{n}} \qquad \text{for } n \geq d$$

- For heavy-tailed distributions X, $n(d) \sim d \log d$ is enough to estimate the covariance matrix

Proof

- Several proofs. The nicest is the Ashlwede-Winter's or better Tropp 2010 approach.
- We have that

$$\hat{\Sigma} = \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{X} = \frac{1}{n}\sum_{i=1}^{n} X_i X_i^\top$$

  is a sum of independent random matrices $X_i X_i^\top$.
- Use matrix versions of the classical scalar deviation inequalities (Chernoff, Hoeffding, Bernstein, Bennett, etc.) for sums of random *matrices*
- These inequalities can be derived almost in the *same* fashion as for *scalar* random variables

Bernstein's inequality for random matrices

Theorem [Matrix Bernstein's inequality]
Let $A_1, \ldots, A_n$ be symmetric $d \times d$ independent matrices such that

$$\mathbb{E}A_i = 0 \quad \text{and} \quad \|A_i\|_{\text{op}} \leq L$$

Put

$$S = \sum_{i=1}^{n} A_i \quad \text{and} \quad \sigma^2 = \|\sum_i \mathbb{E}A_i^2\|_{\text{op}}$$

Then

$$\mathbb{P}[\|S\|_{\text{op}} \geq t] \leq \exp\Big(-\frac{t^2/2}{\sigma^2 + Lt/3}\Big)$$

for any $t \geq 0$ and

$$\mathbb{E}\|S\|_{\text{op}} \leq \sqrt{2\sigma^2 \log(2d)} + \frac{L\log(2d)}{3}$$

[Proof later?]

Consequence for the covariance matrix

$$\Sigma = \mathbb{E}[XX^\top] \qquad \hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} X_i X_i^\top$$

Assume that $\|X\|_2^2 \leq b$ (can be weakened)

- Use Matrix Bernstein with

$$A_i = \frac{1}{n}(X_i X_i^\top - \Sigma)$$

- We have

$$\begin{aligned}
\|A_i\|_{\mathrm{op}} &\leq \frac{1}{n}(\|X_i X_i^\top\|_{\mathrm{op}} + \|\Sigma\|_{\mathrm{op}}) \\
&\leq \frac{1}{n}(\|X_i\|_2^2 + \|\mathbb{E}(XX^\top)\|_{\mathrm{op}}) \\
&\leq \frac{1}{n}(b + \mathbb{E}\|XX^\top\|_{\mathrm{op}}) \leq \frac{2b}{n}
\end{aligned}$$

Bernstein's inequality for random matrices

and for $\sigma^2 = \|\sum_i \mathbb{E}A_i^2\|_{\mathrm{op}}$

$$\begin{aligned}
\mathbb{E}A_i^2 &= \frac{1}{n^2}\mathbb{E}(X_iX_i^\top - \Sigma)^2 \\
&= \frac{1}{n^2}\mathbb{E}\Big(X_iX_i^\top X_iX_i^\top - \Sigma X_iX_i^\top - X_iX_i^\top\Sigma + \Sigma^2\Big) \\
&= \frac{1}{n^2}\mathbb{E}\Big(\|X_i\|_2^2 X_iX_i^\top - \Sigma^2 - \Sigma^2 + \Sigma^2\Big) \\
&\preceq \frac{1}{n^2}\Big(b\Sigma - \Sigma^2\Big) \\
&\preceq \frac{1}{n^2}b\Sigma
\end{aligned}$$

so that

$$\sigma^2 \preceq \frac{b}{n}\|\Sigma\|_{\mathrm{op}}$$

Bernstein's inequality for random matrices

Now, using Matrix Bernstein's with $L = \frac{2b}{n}$ we get

$$\mathbb{E}\|\hat{\Sigma} - \Sigma\|_{\mathrm{op}} \leq \sqrt{\frac{2b\|\Sigma\|_{\mathrm{op}} \log(2d)}{n}} + \frac{2b\log(2d)}{3n}$$

This means that

$$n \geq \frac{2b\log(2d)}{\varepsilon^2\|\Sigma\|} \quad \text{entails} \quad \mathbb{E}\|\hat{\Sigma} - \Sigma\|_{\mathrm{op}} \leq (\varepsilon + \varepsilon^2)\|\Sigma\|_{\mathrm{op}}$$

We often have $b = c \cdot d$ so that

$$n \approx \varepsilon^{-2} d \log d$$

samples are necessary for covariance estimation using $\hat{\Sigma}$

- Sharp for worst-case distribution

Bernstein's inequality for random matrices

[Proof of Matrix Bernstein's inequality]

1. Covariance estimation
   - Estimation of covariance matrices
   - Random matrices with independent entries
   - Random matrices with independent rows
   - Bernstein's inequality for random matrices

2. Inverse covariance / Graphical Gaussian Model
   - A glimpse of graphical modelling / graph theory
   - Graphical Gaussian Model
   - Sparse estimation

3. Some tools from convex optimization
   - Some tools
   - Proximal gradient descent

The Snow storm/Snowmen/Traffic jam example in Paris

- We observe both huge traffic jams and lot of snowmen in Paris
- Also, there's a snow storm in Paris
- There's a strong correlation between the size of the jam and the number of snowmen

Conditional dependencies

- describe better these relationships
- Conditionally on the snow storm, size of traffic jam and number of snowmen are independent

Graphical model: a graph that encodes conditional dependence between coordinates of $(X^1, \dots, X^d) \in \mathbb{R}^d$

Goal

- Learn the graph from a i.i.d sample $X_1, \dots, X_n$ of X
- Stack X_i as lines in $\boldsymbol{X} \in \mathbb{R}^{n \times d}$
- Recall that $X \perp Y | Z$ iff $\mathbb{P}_{(X,Y)|Z} = \mathbb{P}_{X|Z} \otimes \mathbb{P}_{Y|Z}$

Non-directed graphs G

- Nodes $V = \{1, \dots, d\}$
- For $j, k \in V$, $j \sim k$ means that there's an edge between j and k
- Neighbours of $j \in V$

$$\mathrm{ne}(j) = \{k : k \sim j\} \quad \text{and} \quad \mathrm{cl}(j) = \mathrm{ne}(j) \cup \{j\}$$

Non-directed Graphical Model

- Distribution of $X = (X^1, \ldots, X^d)$ is a graphical model according to G if

$$X^j \perp (X^k : k \notin \mathrm{cl}(j)) \mid (X^l : l \in \mathrm{ne}(j))$$

- We write $\mathcal{L}(X) \sim G$ in this case

Unicity

- Note that if $\mathcal{L}(X) \sim G$ and $\mathcal{L}(X) \sim G'$ then $\mathcal{L}(X) \sim G'$
- The graph is *not* unique
- We are interested in the minimal (for inclusion) graph G^* such that $\mathcal{L}(X) \sim G^*$
- Exists and unique if $f_X \ll \mu$ where μ is σ-finite [Lauritzen 96]
- We will call *graph of* X the minimal graph G^* s.t. $\mathcal{L}(X) \sim G^*$

Graphical Gaussian Model

- A Graphical Model with $X \sim N(0, \Sigma)$
- $\Sigma \succ 0$ unknown
- Estimate G^* s.t. $\mathcal{L}(X) \sim G^*$ based on i.i.d copies $X_1, \ldots, X_n$ of X
- What if $n \gg d$ and $n \ll d$?

Once again

$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} X_i X_i^\top = \frac{1}{n} \boldsymbol{X}^\top \boldsymbol{X}$$

Precision matrix and linear regression

- When $X \sim N(0, \Sigma)$ with $\Sigma \succ 0$, G^* is encoded in the *precision matrix* $K = \Sigma^{-1}$
- Defining G with $j \sim k$ iff $K_{j,k} = 0$ gives the minimal graph such that $\mathcal{L}(X) \sim G$
- One of the numerous reasons why we like Gaussian vectors

Lemma

- G defined like that satisfies $\mathcal{L}(X) \sim G$ and is minimal
- For any $j \in V$ there is $\varepsilon_i \sim N(0, K_{j,j}^{-1})$ satisfying $\varepsilon_j \perp (X^k : k \neq j)$ such that

$$X^j = -\sum_{k \in \mathrm{ne}(j)} \frac{K_{j,k}}{K_{j,j}} X^k + \varepsilon_j$$

Multiple testing approach

- One can prove that (partial correlation)

$$\kappa_{j,k} = \rho(X^j, X^k | X^l : l \neq j, k) = -\frac{K_{j,k}}{\sqrt{K_{j,j} K_{k,k}}}$$

- Hence $j \sim k$ iff $\kappa_{j,k} \neq 0$
- Construct tests with null $\kappa_{j,k} = 0$

Assume $n > d - 2$. Put $\hat{K} = (\hat{\Sigma})^{-1}$ and introduce

$$\hat{\kappa}_{j,k} = -\frac{\hat{K}_{j,k}}{\sqrt{\hat{K}_{j,j} \hat{K}_{k,k}}}$$

Graphical Gaussian Model

It turns out that if $\kappa_{j,k} = 0$ we have [Anderson 84, Chap 4.3]

$$\hat{t}_{j,k} = \sqrt{n-2-d}\frac{\hat{\kappa}_{j,k}}{\sqrt{1-\hat{\kappa}_{j,k}}} \sim \text{Student}(n-d-2)$$

- Compute p-values $(\hat{p}_{j,k} : j < k)$ based on this
- Use a multiple testing procedure [Benjamini-Hochberg 1996]

Leads to a control of the FDR (False Discovery Rate)

- Nice strategy, but requires $n \gg d$, otherwise $\hat{\kappa}_{j,k}$ are pretty bad

Sparse estimation of K

- Improve $\hat{K} = (\hat{\Sigma})^{-1}$ using a sparsity assumption
- Use penalization of the goodness-of-fit
- Convex relaxation of sparsity $= \ell_1$ norm a.k.a "Lasso" (Tishirani 94, Candes 96, etc.)
- We want to use

$$\text{pen}(K) = \lambda \sum_{j \neq k} |K_{j,k}|$$

where $\lambda > 0$ is a level of regularization

Hum... What?

- This deserves further explanations...

Graphical Gaussian Model – Sparse estimation

Let $-\ell(K)$ be the minus log-likelihood of the model (more later...)

- Tempting to use

$$\hat{K} \in \operatorname*{argmin}_{K \succeq 0} \Big\{ -\ell(K) + \lambda \|K\|_0 \Big\},$$

  where

$$\|K\|_0 = \#\{(j,k) : K_{j,k} \neq 0\}$$

- But, to do it exactly, you need to try **all** possible subsets of non-zero coordinates of K: 2^{2d} possibilities. Impossible!

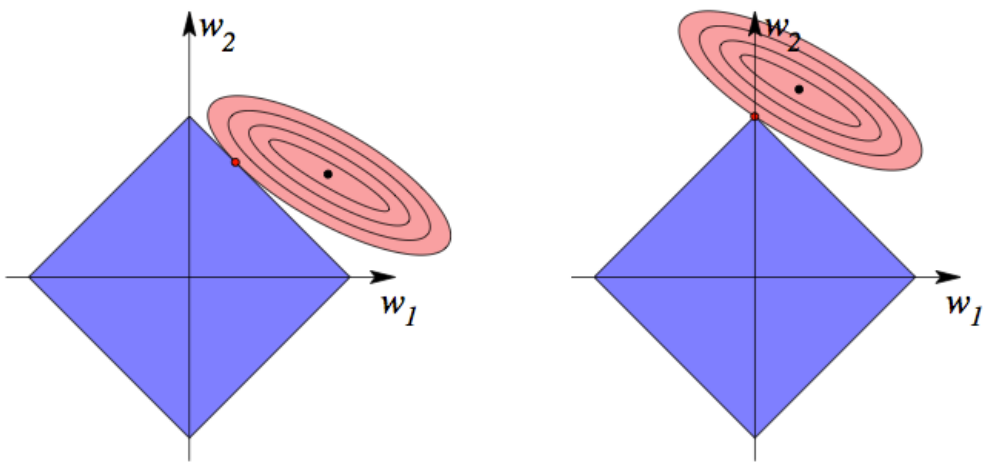
[Link with model-selection, AIC, BIC]

A solution: **Lasso** penalization (least absolute shrinkage and selection operator)

$$\text{pen}(K) = \lambda \|K\|_1 = \sum_{j<k} |K_{j,k}|.$$

This is penalization based on the ℓ_1-norm $\|\cdot\|_1$.

- Why do ℓ_1-penalization leads to sparsity?



Why ℓ_2^2 (ridge) does not induce sparsity?

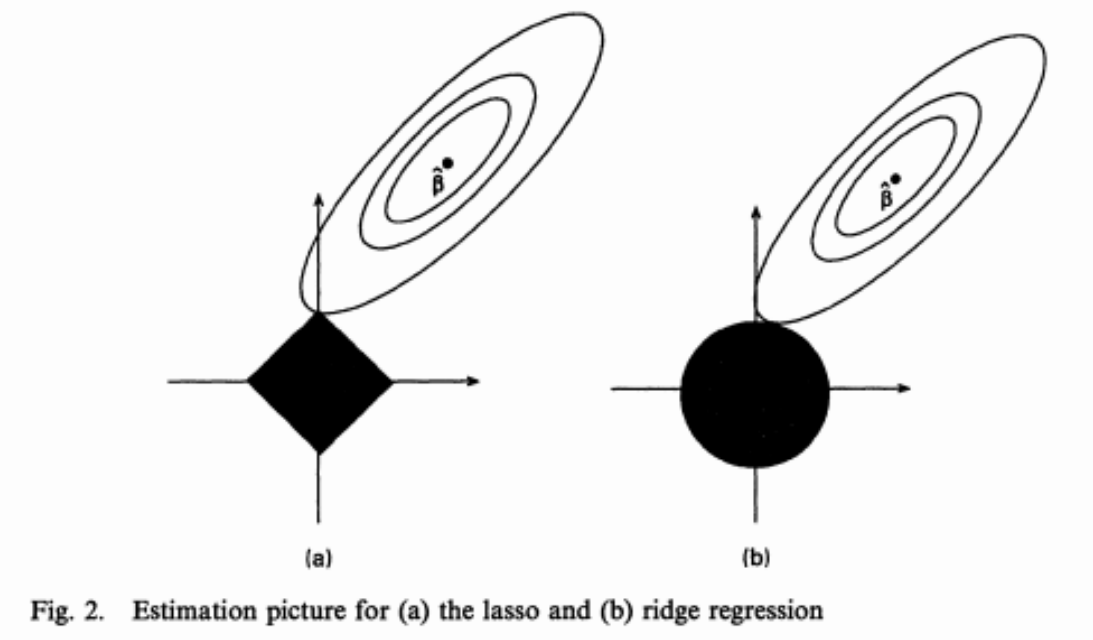


Fig. 2. Estimation picture for (a) the lasso and (b) ridge regression

Consider the minimization problem

$$\min_{a \in \mathbb{R}} \frac{1}{2}(a-b)^2 + \lambda |a|$$

for $\lambda > 0$ and $b \in \mathbb{R}$

- Derivative at 0_+: $d_+ = \lambda - b$
- Derivative at 0_-: $d_- = -\lambda - b$

Let a_* be the solution

- $a_* = 0$ iff $d_+ \geq 0$ and $d_- \leq 0$, namely $|b| \leq \lambda$
- $a_* \geq 0$ iff $d_+ \leq 0$, namely $b \geq \lambda$ and $a_* = b - \lambda$
- $a_* \leq 0$ iff $d_- \geq 0$, namely $b \leq -\lambda$ and $a_* = b + \lambda$

Hence

$$a_* = \text{sign}(b)(|b| - \lambda)_+$$

where $a_+ = \max(0, a)$

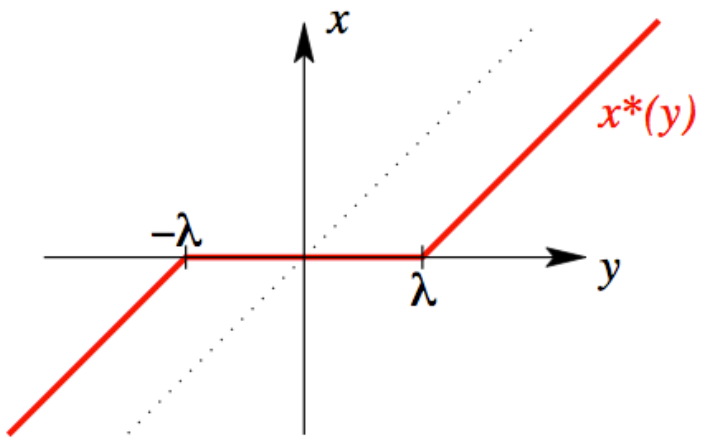
As a consequence, we have

$$A_* = \underset{A \in \mathbb{R}^{d \times d}}{\operatorname{argmin}} \frac{1}{2}\|A - B\|_F^2 + \lambda \|A\|_1 = S_\lambda(B)$$

where

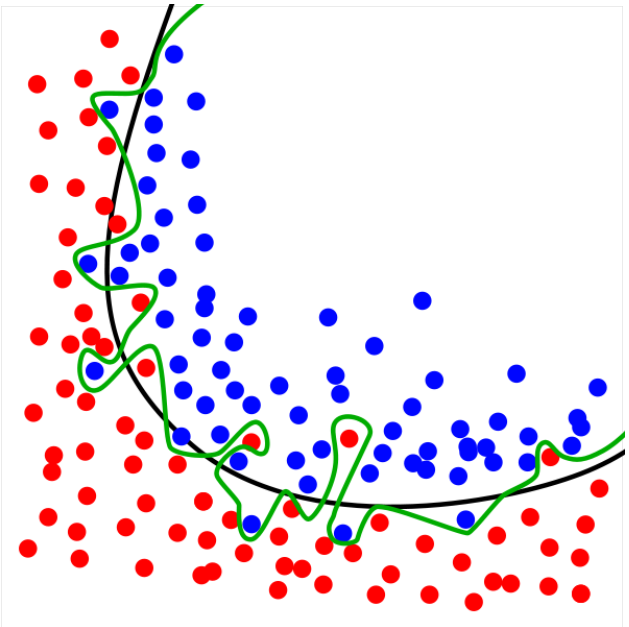
$$S_\lambda(B) = \operatorname{sign}(B) \odot (|B| - \lambda)_+ \quad \text{or} \quad (S_\lambda(B))_{j,k} = \operatorname{sign}(B_{j,k})(|B_{j,k}| - \lambda)_+$$

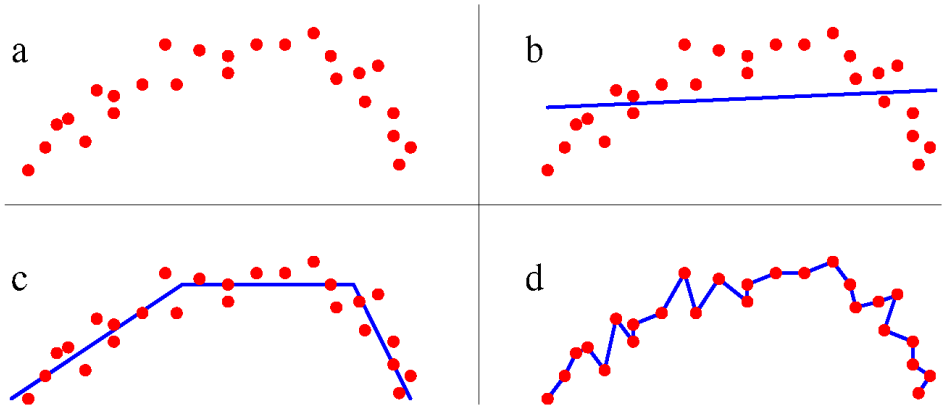
is the **soft-thresholding** operator



A take-home message

- More generally it is a good idea to *penalize*
- Avoid overfitting
- And use cross-validation!





Likelihood of $X_1, \ldots, X_n$ i.i.d. $N(0, \Sigma)$ writes

$$\prod_{i=1}^{n} \frac{1}{(2\pi)^{p/2}\sqrt{\det \Sigma}} \exp(-\frac{1}{2} X_i^\top \Sigma^{-1} X_i)$$

so that using $K = \Sigma^{-1}$ gives

$$\Big(\frac{\det K}{(2\pi)^d}\Big)^{n/2} \exp\Big(-\frac{1}{2}\sum_{i=1}^{n} X_i^\top K X_i\Big)$$

and noting $X_i^\top K X_i = \mathrm{tr}(X_i^\top K X_i) = \mathrm{tr}(X_i X_i^\top K) = \langle X_i X_i^\top, K\rangle$ we arrive at

$$\text{Likelihood}(K) = \Big(\frac{\det K}{(2\pi)^d}\Big)^{n/2} \exp\Big(-\frac{n}{2}\langle \hat{\Sigma}, K\rangle\Big)$$

(recall that $\langle A, B\rangle = \mathrm{tr}(A^\top B)$)

So that the minus log-likelihood writes (forgetting constant terms)

$$-\ell(K) = -\log\det K + \langle \hat{\Sigma}, K \rangle$$

Graphical Lasso [Friedman et al (2007), Banerjee et al (2008)]

$$\hat{K}_\lambda \in \underset{K : K \succ 0}{\operatorname{argmin}} \left\{ -\log\det K + \langle \hat{\Sigma}, K \rangle + \lambda \sum_{1 \leq j < k \leq p} |K_{j,k}| \right\}$$

It is a convex problem!

`Proof.`
Only need to prove that $K \mapsto -\log\det K$ on the SDP cone

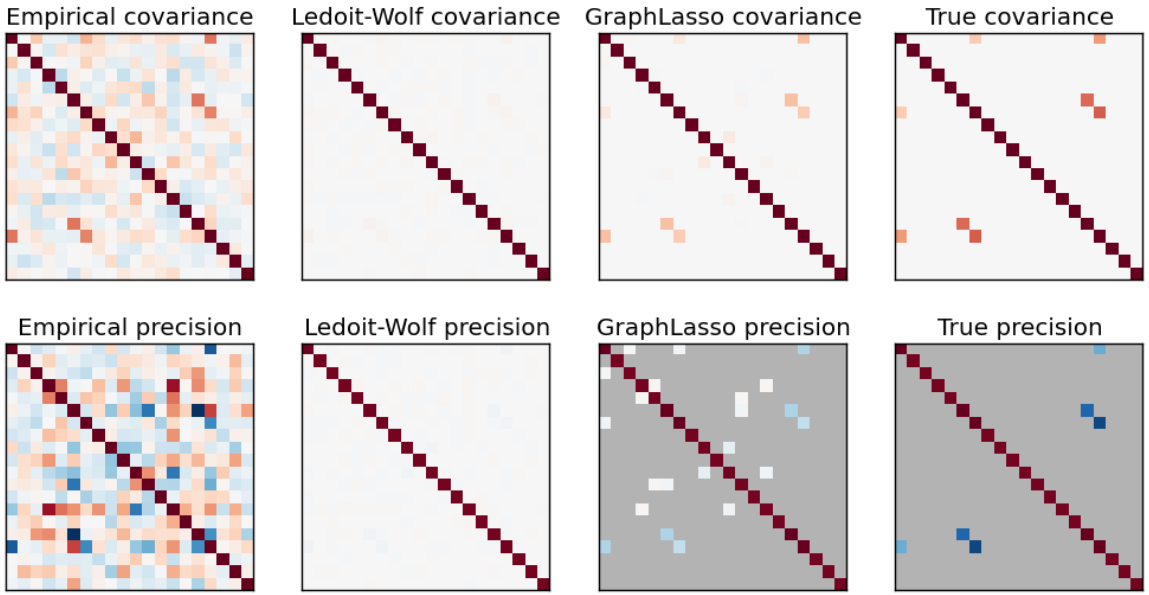
Take $K_1, K_2 \succ 0$ and $\alpha \in [0,1]$
Consider $K_1^{-1/2} K_2 K_1^{-1/2} = UDU^\top$ with $U^\top U = I$ and take $Q = K_1^{1/2} U$
Note that $QQ^\top = K_1$ and $QDQ^\top = K_2$. Then

$$
\begin{aligned}
&-\log\det(\alpha K_1 + (1-\alpha)K_2) \\
&\qquad = -\log\det(Q(\alpha I + (1-\alpha)D)Q^\top) \\
&\qquad = -\log\det(QQ^\top) - \log\det(\alpha I + (1-\alpha)D) \\
&\qquad = -\log\det(QQ^\top) - \log(\alpha + (1-\alpha)\det D) \\
&\qquad \le -\log\det(QQ^\top) - \alpha\log(1) - (1-\alpha)\log\det(D) \\
&\qquad = -\alpha\log\det(QQ^\top) - (1-\alpha)\log\det(QQ^\top) \\
&\qquad\quad - (1-\alpha)\log\det(D) \\
&\qquad = -\alpha\log\det(QQ^\top) - (1-\alpha)\log\det(QDQ^\top)
\end{aligned}
$$

where we used the convexity of $x \mapsto -\log x$ and the fact that $\det(AB) = \det(BA) = \det(A)\det(B)$.

Graphical Gaussian Model – Sparse estimation

- Nice optimization routines (block coordinate descent) [see Friedman et al. 07]
- `R` package `gLasso`
- Featured in Python's `sklearn` as well
- But when $n \leq d$ leads sometimes to unsatisfactory results compared to other approaches



Estimation by regression

- Recall that

$$X^j = -\sum_{k \in \mathrm{ne}(j)} \frac{K_{j,k}}{K_{j,j}} X^k + \varepsilon_j$$

  for any $j \in V$ where $\varepsilon_i \sim N(0, K_{j,j}^{-1})$ and $\varepsilon_j \perp (X^k : k \neq j)$

- Put

$$\theta_{j,k} = -\frac{K_{j,k}}{K_{k,k}} \quad \text{if} \quad k \neq j \quad \text{and} \quad \theta_{j,j} = 0 \quad \text{otherwise}$$

- This gives

$$\mathbb{E}(X^j | X^k : k \neq j) = \sum_k \theta_{k,j} X^k$$

So it's tempting to minimize

$$(\theta_{k,j})_{k\neq j} \in \operatorname*{argmin}_{\beta\in\mathbb{R}^{d-1}} \mathbb{E}\Big[\Big(X^j - \sum_{k\neq j}\beta_k X^k\Big)^2\Big]$$

Consider Θ as the set of $d \times d$ matrices with zeros on the diagonal. Summing over j leads to

$$\begin{aligned}\theta &\in \operatorname*{argmin}_{\theta\in\Theta} \mathbb{E}\Big[\sum_j \Big(X^j - \sum_{k\neq j}\theta_{k,j}X^k\Big)^2\Big] \\ &= \operatorname*{argmin}_{\theta\in\Theta} \mathbb{E}\|X - \theta^\top X\|_2^2\end{aligned}$$

But note that

$$\begin{aligned}
\mathbb{E}\|(I-\theta)^\top X\|_2^2 &= \mathbb{E}[X^\top (I-\theta)(I-\theta)^\top X] \\
&= \mathbb{E}\,\mathrm{tr}(XX^\top (I-\theta)(I-\theta)^\top) \\
&= \mathrm{tr}(\Sigma(I-\theta)(I-\theta)^\top) \\
&= \mathrm{tr}((I-\theta)^\top \Sigma (I-\theta)) \\
&= \|\Sigma^{1/2}(I-\theta)\|_F^2
\end{aligned}$$

so that

$$\theta \in \operatorname*{argmin}_{\theta\in\Theta} \|\Sigma^{1/2}(I-\theta)\|_F^2$$

Now, replace Σ by $\hat{\Sigma}$ to get

$$\begin{aligned}\|\hat{\Sigma}^{1/2}(I-\theta)\|_F^2 &= \frac{1}{n}\langle I-\theta, \boldsymbol{X}^\top \boldsymbol{X}(I-\theta)\rangle \\ &= \frac{1}{n}\|\boldsymbol{X}-\boldsymbol{X}\theta\|_F^2\end{aligned}$$

Leads to another approch:

$$\hat{\theta}_\lambda \in \underset{\theta\in\Theta}{\operatorname{argmin}} \left\{ \frac{1}{n}\|\boldsymbol{X}-\boldsymbol{X}\theta\|_F^2 + \lambda \sum_{j\neq k} |\theta_{j,k}| \right\}$$

A problem with ℓ_1-penalization

- We can have $(\hat{\theta}_\lambda)_{j,k} = 0$ while $(\hat{\theta}_\lambda)_{k,j} \neq 0$ (and conversely)
- What to do then?
- Or VS And condition to decide if $j \sim k$

A solution

- Group Lasso: use instead

$$\hat{\theta}_\lambda \in \underset{\theta \in \Theta}{\operatorname{argmin}} \left\{ \frac{1}{n} \|\boldsymbol{X} - \boldsymbol{X}\theta\|_F^2 + \lambda \sum_{j \neq k} \sqrt{\theta_{j,k}^2 + \theta_{k,j}^2} \right\}$$

- Indeed for any $b \in \mathbb{R}^p$ and $\lambda > 0$

$$\underset{a \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ \frac{1}{2} \|a - b\|_2^2 + \lambda \|a\|_2 \right\} = \left(1 - \frac{\lambda}{\|b\|_2}\right)_+ b$$

(group soft-thresholding)

Theoretical guarantees

Under technical conditions, one gets

$$\|\Sigma^{1/2}(\hat{\theta}_\lambda - \theta)\|_F^2 \leq c\frac{\log d}{n}\|\theta_0\|$$

with a large probability [Giraud 2014 Thm 7.3]

How to compute $\hat{\theta}_\lambda$?

- Convex optimization
- Convex smooth + convex non-smooth problem
- Simplest algorithm: proximal gradient descent

1. Covariance estimation
   - Estimation of covariance matrices
   - Random matrices with independent entries
   - Random matrices with independent rows
   - Bernstein's inequality for random matrices

2. Inverse covariance / Graphical Gaussian Model
   - A glimpse of graphical modelling / graph theory
   - Graphical Gaussian Model
   - Sparse estimation

3. Some tools from convex optimization
   - Some tools
   - Proximal gradient descent

Some tools from convex optimization

We want to solve a problem of the form

$$\hat{\theta} \in \operatorname*{argmin}_{\theta} \{f(\theta) + g(\theta)\}$$

where

- f is convex and smooth
- g is convex but not smooth

Example

- $f(\theta) = \frac{1}{2n}\|\boldsymbol{X} - \boldsymbol{X}\theta\|_F^2$
- $g(\theta) = \lambda\|\theta\|_1$

Some tools from convex optimization

- For any g convex [lower semi-continuous] and any $y \in \mathbb{R}^d$, we define the **proximal operator**

$$\text{prox}_g(y) = \underset{x \in \mathbb{R}^d}{\text{argmin}} \left\{ \frac{1}{2}\|x - y\|_2^2 + g(x) \right\}$$

  (strongly convex problem $\Rightarrow$ unique minimum)

- We already proved that soft-thresholding is the proximal operator of the ℓ_1-norm

$$\text{prox}_{\lambda\|\cdot\|_1}(y) = S_\lambda(y) = \text{sign}(y) \odot (|y| - \lambda)+$$

Proximal operators and proximal algorithms are important tools for optimization in machine learning

Some tools from convex optimization

- $g(x) = c$ for a constant c, $\text{prox}_g = Id$
- If C convex set, and

$$g(x) = \delta_C(x) = \begin{cases} 0 & \text{if } x \in C \\ +\infty & \text{if } x \notin C \end{cases}$$

then

$$\text{prox}_g = \text{proj}_C = \text{projection onto } C.$$

- If $g(x) = \frac{1}{2}\|x\|_2^2$ then

$$\text{prox}_{\lambda g}(x) = \frac{1}{1+\lambda} x = \text{shrinkage operator}$$

- If $g(x) = \|x\|_2$ then

$$\text{prox}_{\lambda g}(x) = \left(1 - \frac{\lambda}{\|x\|_2}\right)_+ x,$$

the block soft-thresholding operator

- If $g(x) = \|x\|_1 + \frac{\gamma}{2}\|x\|_2^2$ (elastic-net) where $\gamma > 0$, then

$$\text{prox}_{\lambda g}(x) = \frac{1}{1+\lambda\gamma}\,\text{prox}_{\lambda\|\cdot\|_1}(x)$$

  [elastic-net and features correlation]

- If $g(x) = \sum_{p\in\mathcal{P}} \|x_p\|_2$ where $\mathcal{P}$ partition of $\{1,\dots,d\}$,

$$(\text{prox}_{\lambda g}(x))_p = \left(1 - \frac{\lambda}{\|x_p\|_2}\right)_+ x_p,$$

  for $p \in \mathcal{P}$. Block soft-thresholding, used for group-Lasso

Some tools from convex optimization

Let $f : \mathbb{R}^d \to [-\infty, +\infty]$ be a convex function

- f is L-**smooth** if it is continuously differentiable and if

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2 \quad \text{for any } x, y \in \mathbb{R}^d$$

  In this case we have the **descent lemma**

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2$$

  for all x, y, and if f twice continuously differentiable then $H_f(x) \preceq L I_d$, where $H_f(x)$ Hessian at x of f

- f is μ-strongly convex if $f(\cdot) - \frac{\mu}{2}\|\cdot\|_2^2$ is convex. Equivalent to

$$f(y) \geq f(x) + \langle g, y - x \rangle + \frac{\mu}{2}\|y - x\|_2^2$$

  for $g \in \partial f(x)$. Equivalent to $H_f(x) \succeq \mu I_d$ when twice differentiable.

1. Covariance estimation
   - Estimation of covariance matrices
   - Random matrices with independent entries
   - Random matrices with independent rows
   - Bernstein's inequality for random matrices

2. Inverse covariance / Graphical Gaussian Model
   - A glimpse of graphical modelling / graph theory
   - Graphical Gaussian Model
   - Sparse estimation

3. Some tools from convex optimization
   - Some tools
   - Proximal gradient descent

Proximal gradient descent

Now, how to solve

$$\hat{\theta} \in \operatorname*{argmin}_{\theta} \{f(\theta) + g(\theta)\}$$

where

- f is convex and L-smooth
- g is convex and continuous, but possibly non-smooth (for instance ℓ_1 penalization)
- g is **prox-capable**: not hard to compute its proximal operator

Example

- $f(\theta) = \frac{1}{2n}\|\boldsymbol{X} - \boldsymbol{X}\theta\|_F^2$, $\nabla f(\theta) = \frac{1}{n}\boldsymbol{X}^\top(\boldsymbol{X}\theta - \boldsymbol{X})$ and $L = \frac{1}{n}\|\boldsymbol{X}^\top\boldsymbol{X}\|_{\text{op}}$
- $g(\theta) = \lambda\|\theta\|_1$, prox_g is easy to compute (soft-thresholding)

Proximal gradient descent

Now how do I minimize $f + g$?

- Key point: the **descent lemma**. If f convex and L-smooth, then for any $L' \geq L$:

$$f(\theta') \leq f(\theta) + \langle \nabla f(\theta), \theta' - \theta \rangle + \frac{L'}{2} \|\theta' - \theta\|_2^2$$

  for any $\theta, \theta' \in \mathbb{R}^d$

- At iteration k, the current point is θ^k. I use the descent lemma:

$$f(\theta) \leq f(\theta^k) + \langle \nabla f(\theta^k), \theta - \theta^k \rangle + \frac{L'}{2} \|\theta - \theta^k\|_2^2.$$

Proximal gradient descent

- Remark that

$$\begin{aligned}
&\operatorname*{argmin}_{\theta\in\mathbb{R}^d}\left\{f(\theta^k)+\langle\nabla f(\theta^k),\theta-\theta^k\rangle+\frac{L'}{2}\|\theta-\theta^k\|_2^2\right\}\\
&\quad=\operatorname*{argmin}_{\theta\in\mathbb{R}^d}\left\|\theta-\left(\theta^k-\frac{1}{L'}\nabla f(\theta^k)\right)\right\|_2^2
\end{aligned}$$

- Hence, choose

$$\theta^{k+1}=\theta^k-\frac{1}{L'}\nabla f(\theta^k)$$

  This is the basic **gradient descent** algorithm
- Gradient descent is based on a **majoration-minimization** principle, with a quadratic majorant given by the descent lemma
- But we forgot about g...

Proximal gradient descent

Let's put back g:

$$f(\theta) + g(\theta) \leq f(\theta^k) + \langle \nabla f(\theta^k), \theta - \theta^k \rangle + \frac{L'}{2} \|\theta - \theta^k\|_2^2 + g(\theta)$$

and again

$$\begin{aligned}
&\operatorname*{argmin}_{\theta \in \mathbb{R}^d} \left\{ f(\theta^k) + \langle \nabla f(\theta^k), \theta - \theta^k \rangle + \frac{L'}{2} \|\theta - \theta^k\|_2^2 + g(\theta) \right\} \\
&\quad = \operatorname*{argmin}_{\theta \in \mathbb{R}^d} \left\{ \frac{L'}{2} \left\| \theta - \left(\theta^k - \frac{1}{L'} \nabla f(\theta^k) \right) \right\|_2^2 + g(\theta) \right\} \\
&\quad = \operatorname*{argmin}_{\theta \in \mathbb{R}^d} \left\{ \frac{1}{2} \left\| \theta - \left(\theta^k - \frac{1}{L'} \nabla f(\theta^k) \right) \right\|_2^2 + \frac{1}{L'} g(\theta) \right\} \\
&\quad = \operatorname{prox}_{g/L'} \left(\theta^k - \frac{1}{L'} \nabla f(\theta^k) \right)
\end{aligned}$$

The prox operator naturally appears because of the descent lemma

Proximal gradient descent

Proximal gradient descent algorithm [also called ISTA]

- **Input**: starting point θ^0, Lipschitz constant $L > 0$ for ∇f
- For $k = 1, 2, \ldots$ until *converged* do
  - $\theta^k = \text{prox}_{g/L}\left(\theta^{k-1} - \frac{1}{L}\nabla f(\theta^{k-1})\right)$
- **Return** last θ^k

Also called **Forward-Backward splitting**. For regression-based GGM, iteration is

$$\theta^k = S_{\lambda/L}\left(\theta^{k-1} - \frac{1}{L}(\boldsymbol{X}^\top \boldsymbol{X}\theta^{k-1} - \boldsymbol{X}^\top \boldsymbol{X})\right),$$

where S_λ is the soft-thresholding operator

Proximal gradient descent

- Put for short $F = f + g$,
- Take any $\theta^* \in \operatorname{argmin}_{\theta \in \mathbb{R}^d} F(\theta)$

Theorem (Beck Teboulle (2009))

If the sequence $\{\theta^k\}$ is generated by ISTA, then

$$F(\theta^k) - F(\theta^*) \leq \frac{L\|\theta^0 - \theta^*\|_2^2}{2k}$$

- Convergence rate is $O(1/k)$
- Is it possible to improve the $O(1/k)$ rate?

Yes! Using **Accelerated proximal gradient descent** (called FISTA, Nesterov 83, 04, Beck Teboule 09)

- Idea: to find θ^{k+1}, use an interpolation between θ^k and θ^{k-1}

Accelerated proximal gradient descent algorithm [FISTA]

- **Input**: starting points $z^1 = \theta^0$, Lipschitz constant $L > 0$ for ∇f, $t_1 = 1$
- For $k = 1, 2, \ldots$ until *converged* do
  - $\theta^k = \text{prox}_{g/L}(z^k - \frac{1}{L}\nabla f(z^k))$
  - $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$
  - $z^{k+1} = \theta_k + \frac{t_k - 1}{t_{k+1}}(\theta^k - \theta^{k-1})$
- **Return** last θ^k

Proximal gradient descent

Theorem (Beck Teboulle (2009))

If the sequence $\{\theta^k\}$ is generated by FISTA, then

$$F(\theta^k) - F(\theta^*) \leq \frac{2L\|\theta^0 - \theta^*\|_2^2}{(k+1)^2}$$

- Convergence rate is $O(1/k^2)$
- Is $O(1/k^2)$ the optimal rate in general?

Yes. Put $g = 0$

Theorem (Nesterov)

For any optimization procedure satisfying

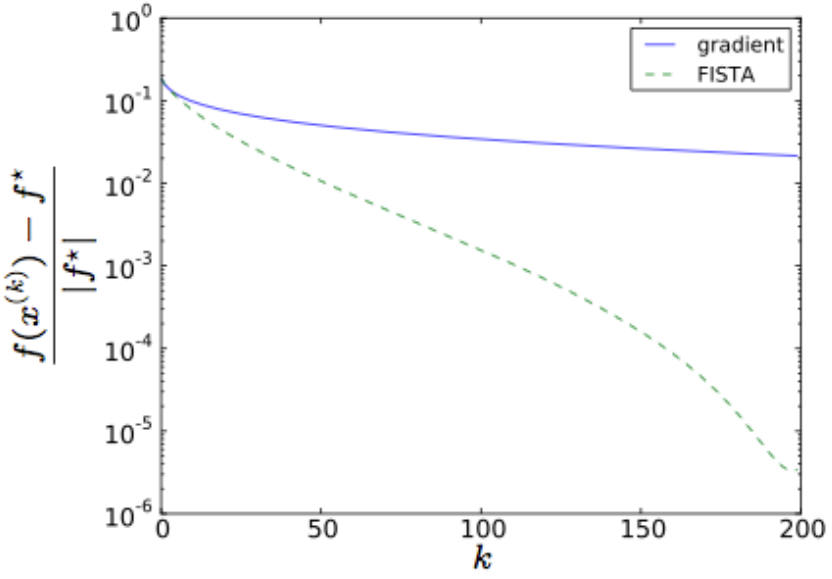
$$\theta^{k+1} \in \theta^1 + \operatorname{span}(\nabla f(\theta^1), \ldots, \nabla f(\theta^k)),$$

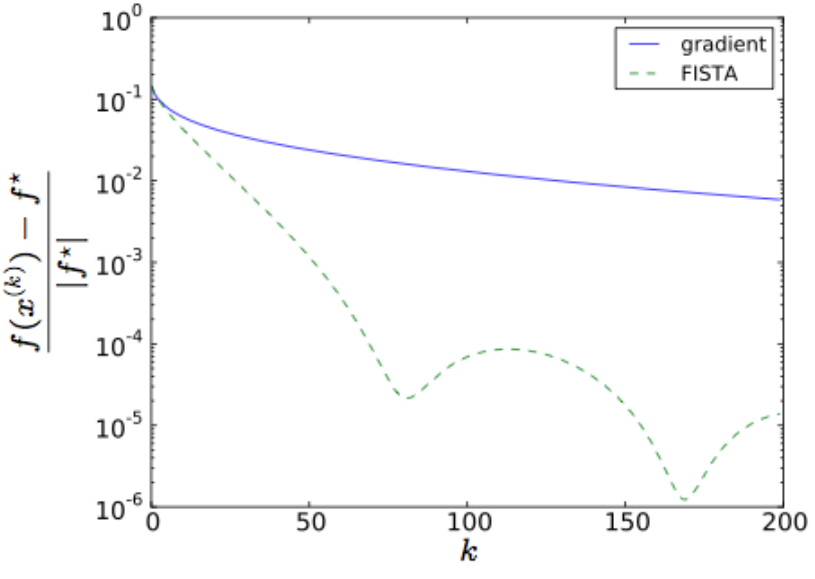
there is a function f on $\mathbb{R}^d$ convex and L-smooth such that

$$\min_{1 \le j \le k} f(\theta^j) - f(\theta^*) \ge \frac{3L}{32} \frac{\|\theta^1 - \theta^*\|_2^2}{(k+1)^2}$$

for any $1 \le k \le (d-1)/2$.

Comparison of ISTA and FISTA





FISTA is **not** a descent algorithm, while ISTA is

Now, if f is L-smooth and μ-strongly convex (we can always use ridge regularization)

Theorem

If f is L-smooth and μ-strongly convex and if the sequence $\{\theta^k\}$ is generated by ISTA, then

$$F(\theta^k) - F(\theta^*) \leq \left(1 - \frac{\mu}{L}\right)^k (F(\theta_0) - F(\theta^*))$$

and if $\{\theta^k\}$ is generated by FISTA, then

$$F(\theta^k) - F(\theta^*) \leq \left(1 - \sqrt{\frac{\mu}{L}}\right)^k (F(\theta_0) - F(\theta^*))$$

Much faster **linear** rates

What if I don't know $L > 0$?

- $\|\boldsymbol{X}^\top \boldsymbol{X}\|_{\text{op}}$ can be long to compute
- Letting L evolve along iterations k generally improve convergence speed

Backtracking linesearch. Idea:

- Start from a very small lipschitz constant L
- Between iteration k and $k+1$, choose the smallest L satisfying the lemma descent at z^k

Proximal gradient descent

At iteration k of FISTA, we have z^k and a constant L_k

1. Put $L \leftarrow L_k$
2. Do an iteration

$$\theta \leftarrow \text{prox}_{g/L}\left(z^k - \frac{1}{L}\nabla f(z^k)\right)$$

3. Check it this step satisfies the descent lemma at z^k:

$$f(\theta) + g(\theta) \leq f(z^k) + \langle \nabla f(z^k), \theta - z^k \rangle + \frac{L}{2}\|\theta - z^k\|_2^2 + g(\theta)$$

4. If yes, then $\theta^{k+1} \leftarrow \theta$ and $L_{k+1} \leftarrow L$ and continue FISTA
5. It not, then put $L \leftarrow 2L$ (say), and go back to point 2

Sequence L_k is non-decreasing: between iteration k and $k+1$, a tweak is to *decrease* it a little bit to have (much) faster convergence

Thank you!