Sparse Estimation for Image and Vision Processing

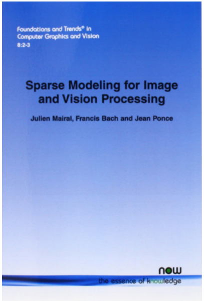
Alberto Bietti and Julien Mairal

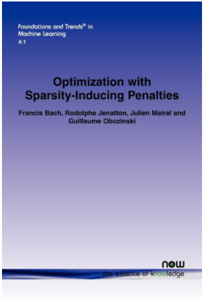
Inria, Grenoble

Autrans, SMAI-MODE, 2018
Part III

Course material (freely available on arXiv)

J. Mairal, F. Bach and J. Ponce. *Sparse Modeling for Image and Vision Processing*. Foundations and Trends in Computer Graphics and Vision. 2014.





F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. *Optimization with sparsity-inducing penalties*. Foundations and Trends in Machine Learning, 4(1). 2012.

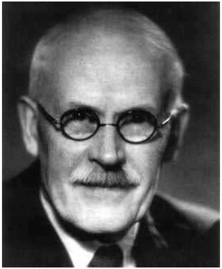
Outline

Part I: A Short Introduction to Parcimony

1. A short introduction to parsimony
   - Early thoughts
   - Sparsity in the statistics literature from the 60's and 70's
   - Wavelet thresholding in signal processing from the 90's
   - The modern parsimony and the ℓ_1-norm
   - Structured sparsity
   - Compressed sensing and sparse recovery
2. Discovering the structure of natural images
3. Sparse models for image processing
4. Optimization for sparse estimation
5. Application cases

Early thoughts

(a) Dorothy Wrinch
1894–1980



(b) Harold Jeffreys
1891–1989

The existence of simple laws is, then, apparently, to be regarded as a quality of nature; and accordingly we may infer that it is justifiable to prefer a simple law to a more complex one that fits our observations slightly better.

[Wrinch and Jeffreys, 1921]. Philosophical Magazine Series.

Chronological overview of parsimony

- 14th century: Ockham's razor;
- 1921: Wrinch and Jeffreys' simplicity principle;
- 1952: Markowitz's portfolio selection;
- 60 and 70's: best subset selection in statistics;
- 70's: use of the ℓ_1-norm for signal recovery in geophysics;
- 90's: wavelet thresholding in signal processing;
- 1996: Olshausen and Field's dictionary learning;
- 1996–1999: Lasso (statistics) and basis pursuit (signal processing);
- 2006: compressed sensing (signal processing) and Lasso consistency (statistics);
- 2006–now: applications of dictionary learning in various scientific fields such as image processing and computer vision.

Sparsity in the statistics literature from the 60's and 70's

Given some observed data points $\mathbf{z}_1, \ldots, \mathbf{z}_n$ that are assumed to be independent samples from a statistical model with parameters $\boldsymbol{\theta}$ in $\mathbb{R}^p$, **maximum likelihood estimation (MLE)** consists of minimizing

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left[\mathcal{L}(\boldsymbol{\theta}) \triangleq -\sum_{i=1}^{n} \log P_{\boldsymbol{\theta}}(\mathbf{z}_i) \right].$$

Example: ordinary least square

Observations $\mathbf{z}_i = (y_i, \mathbf{x}_i)$, with y_i in $\mathbb{R}$.
Linear model: $y_i = \mathbf{x}_i^\top \boldsymbol{\theta} + \varepsilon_i$, with $\varepsilon_i \sim \mathcal{N}(0, 1)$.

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \sum_{i=1}^{n} \frac{1}{2} \left(y_i - \mathbf{x}_i^\top \boldsymbol{\theta} \right)^2.$$

Sparsity in the statistics literature from the 60's and 70's

Given some observed data points $\mathbf{z}_1, \ldots, \mathbf{z}_n$ that are assumed to be independent samples from a statistical model with parameters $\boldsymbol{\theta}$ in $\mathbb{R}^p$, **maximum likelihood estimation (MLE)** consists of minimizing

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left[\mathcal{L}(\boldsymbol{\theta}) \triangleq - \sum_{i=1}^{n} \log P_{\boldsymbol{\theta}}(\mathbf{z}_i) \right].$$

Motivation for finding a sparse solution:

- removing irrelevant variables from the model;
- obtaining an easier interpretation;
- preventing overfitting;

Sparsity in the statistics literature from the 60's and 70's

Given some observed data points $\mathbf{z}_1, \ldots, \mathbf{z}_n$ that are assumed to be independent samples from a statistical model with parameters $\boldsymbol{\theta}$ in $\mathbb{R}^p$, **maximum likelihood estimation (MLE)** consists of minimizing

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left[\mathcal{L}(\boldsymbol{\theta}) \triangleq - \sum_{i=1}^{n} \log P_{\boldsymbol{\theta}}(\mathbf{z}_i) \right].$$

Two questions:

1. how to choose k?
2. how to find the best subset of k variables?

Sparsity in the statistics literature from the 60's and 70's

How to choose k?

- Mallows's C_p statistics [Mallows, 1964, 1966];
- Akaike information criterion (AIC) [Akaike, 1973];
- Bayesian information critertion (BIC) [Schwarz, 1978];
- Minimum description length (MDL) [Rissanen, 1978].

These approaches lead to penalized problems

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \mathcal{L}(\boldsymbol{\theta}) + \lambda \|\boldsymbol{\theta}\|_0,$$

with different choices of λ depending on the chosen criterion.

Sparsity in the statistics literature from the 60's and 70's

How to solve the best k-subset selection problem?

Unfortunately...

...the problem is NP-hard [Natarajan, 1995].

Two strategies

- **combinatorial exploration** with branch-and-bound techniques [Furnival and Wilson, 1974] $\rightarrow$ **leaps and bounds**, exact algorithm but exponential complexity;
- **greedy approach**: forward selection [Efroymson, 1960] (originally developed for observing *intermediate* solutions), already contains all the ideas of **matching pursuit** algorithms.

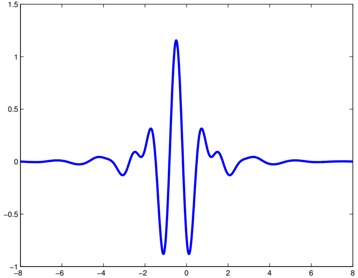
Important reference: [Hocking, 1976]. *The analysis and selection of variables in linear regression*. Biometrics.

Wavelet thresholding in signal processing from the 90's

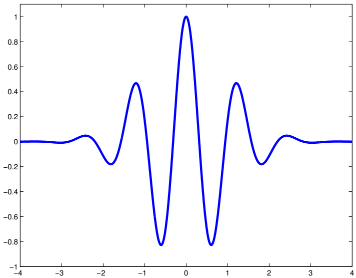
A wavelet basis represents a set of functions φ_1, φ_2 that are essentially dilated and shifted versions of each other [see Mallat, 2008].

Concept of parsimony with wavelets

When a signal f is "smooth", it is close to an expansion $\sum_i \alpha_i \varphi_i$ where only a few coefficients α_i are non-zero.



(a) Meyer's wavelet

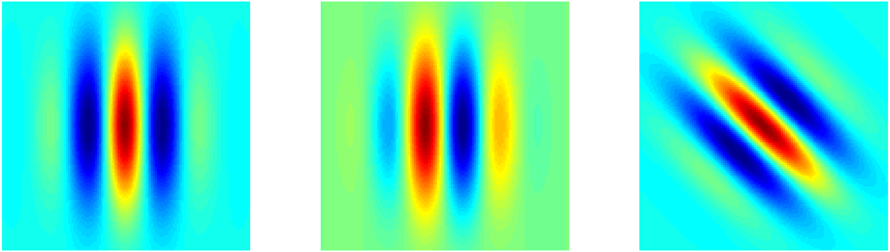


(b) Morlet's wavelet

Wavelet thresholding in signal processing from the 90's

Wavelets where the topic of a long quest for representing natural images

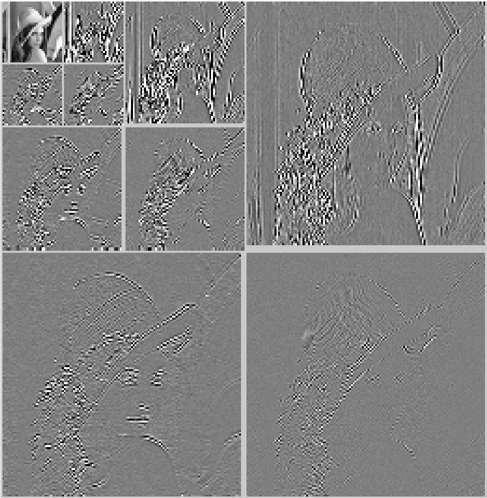
- 2D-Gabors [Daugman, 1985];
- steerable wavelets [Simoncelli et al., 1992];
- curvelets [Candès and Donoho, 2002];
- countourlets [Do and Vertterli, 2003];
- bandlets [Le Pennec and Mallat, 2005];
- $\star$-lets (joke).



(a) 2D Gabor filter. (b) With shifted phase. (c) With rotation.

Wavelet thresholding in signal processing from 90's

The theory of wavelets is well developed for continuous signals, *e.g.*, in $L^2(\mathbb{R})$, but also for discrete signals $\mathbf{x}$ in $\mathbb{R}^n$.



Wavelet thresholding in signal processing from 90's

Given an orthogonal wavelet basis $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_n]$ in $\mathbb{R}^{n \times n}$, the wavelet decomposition of $\mathbf{x}$ in $\mathbb{R}^n$ is simply

$$\boldsymbol{\beta} = \mathbf{D}^\top \mathbf{x} \quad \text{and we have} \quad \mathbf{x} = \mathbf{D}\boldsymbol{\beta}.$$

The k-sparse approximation problem

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\|_0 \leq k,$$

is not NP-hard here: since $\mathbf{D}$ is orthogonal, it is equivalent to

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\boldsymbol{\beta} - \boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\|_0 \leq k.$$

Wavelet thresholding in signal processing from 90's

Given an orthogonal wavelet basis $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_n]$ in $\mathbb{R}^{n \times n}$, the wavelet decomposition of $\mathbf{x}$ in $\mathbb{R}^n$ is simply

$$\boldsymbol{\beta} = \mathbf{D}^\top \mathbf{x} \quad \text{and we have} \quad \mathbf{x} = \mathbf{D}\boldsymbol{\beta}.$$

The k-sparse approximation problem

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\|_0 \leq k,$$

The solution is obtained by **hard-thresholding**:

$$\boldsymbol{\alpha}^{\text{ht}}[j] = \delta_{|\boldsymbol{\beta}[j]| \geq \mu} \boldsymbol{\beta}[j] = \begin{cases} \boldsymbol{\beta}[j] & \text{if } \ |\boldsymbol{\beta}[j]| \geq \mu \\ 0 & \text{otherwise} \end{cases},$$

where μ the k-th largest value among the set $\{|\boldsymbol{\beta}[1]|, \ldots, |\boldsymbol{\beta}[p]|\}$.

Wavelet thresholding in signal processing, 90's

Another key operator is the **soft-thresholding** operator [see Donoho and Johnstone, 1994] :

$$\boldsymbol{\alpha}^{\text{st}}[j] \triangleq \text{sign}(\boldsymbol{\beta}[j]) \max(|\boldsymbol{\beta}[j]| - \lambda, 0) = \begin{cases} \boldsymbol{\beta}[j] - \lambda & \text{if } \boldsymbol{\beta}[j] \geq \lambda \\ \boldsymbol{\beta}[j] + \lambda & \text{if } \boldsymbol{\beta}[j] \leq -\lambda \\ 0 & \text{otherwise} \end{cases},$$

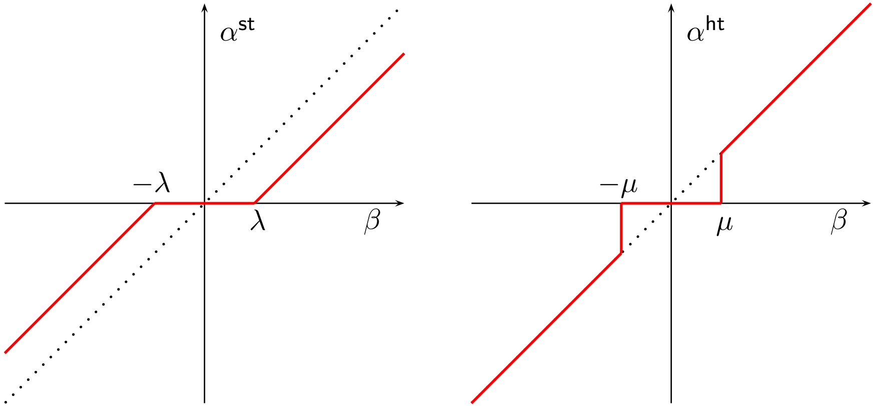
where λ is a parameter playing the same role as μ previously.

With $\boldsymbol{\beta} \triangleq \mathbf{D}^\top \mathbf{x}$ and $\mathbf{D}$ orthogonal, it provides the solution of the following sparse reconstruction problem:

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1,$$

which will be of high importance later.

Wavelet thresholding in signal processing, 90's



(d) Soft-thresholding operator, $\alpha^{\text{st}} = \text{sign}(\beta) \max(|\beta| - \lambda, 0)$.

(e) Hard-thresholding operator $\alpha^{\text{ht}} = \delta_{|\beta| \geq \mu} \beta$.

Figure: Soft- and hard-thresholding operators, which are commonly used for signal estimation with orthogonal wavelet basis.

Wavelet thresholding in signal processing, 90's

Various work tried to exploit the **structure** of wavelet coefficients.

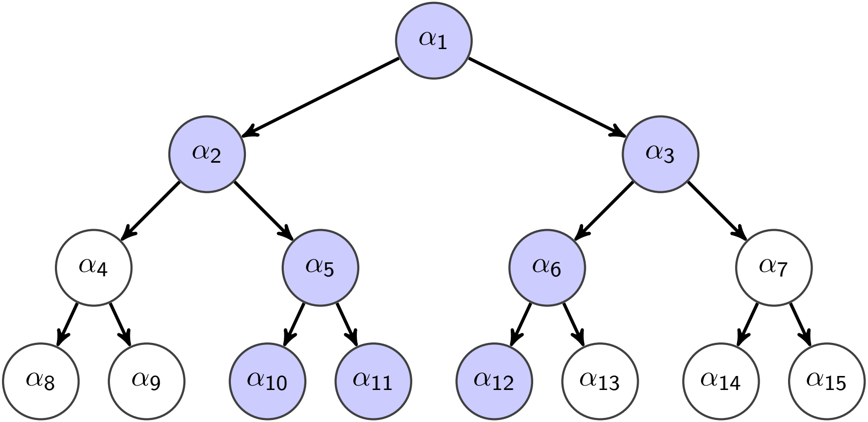


Figure: Illustration of a wavelet tree with four scales for one-dimensional signals. We also illustrate the zero-tree coding scheme [Shapiro, 1993].

Wavelet thresholding in signal processing, 90's

To model spatial relations, it is possible to define some (non-overlapping) groups $\mathcal{G}$ of wavelet coefficients, and define a **group soft-thresholding** operator [Hall et al., 1999, Cai, 1999]. For every group g in $\mathcal{G}$,

$$\boldsymbol{\alpha}^{\text{gt}}[g] \triangleq \begin{cases} \left(1 - \frac{\lambda}{\|\boldsymbol{\beta}[g]\|_2}\right)\boldsymbol{\beta}[g] & \text{if } \|\boldsymbol{\beta}[g]\|_2 \geq \lambda \\ 0 & \text{otherwise} \end{cases},$$

where $\boldsymbol{\beta}[g]$ is the vector of size $|g|$ recording the entries of $\boldsymbol{\beta}$ in g.

With $\boldsymbol{\beta} \triangleq \mathbf{D}^\top \mathbf{x}$ and $\mathbf{D}$ orthogonal, it is in fact the solution of

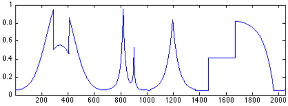
$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \sum_{g \in \mathcal{G}} \|\boldsymbol{\alpha}[g]\|_2,$$

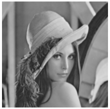
which will be of interest later in the lecture.

The modern parsimony and the ℓ_1-norm

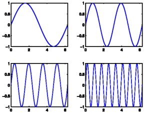
Sparse linear models in signal processing

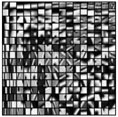
Let $\mathbf{x}$ in $\mathbb{R}^n$ be a signal.





Let $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_p] \in \mathbb{R}^{n \times p}$ be a set of elementary signals.
We call it **dictionary**.





$\mathbf{D}$ is "adapted" to $\mathbf{x}$ if it can represent it with a few elements—that is, there exists a **sparse vector** $\boldsymbol{\alpha}$ in $\mathbb{R}^p$ such that $\mathbf{x} \approx \mathbf{D}\boldsymbol{\alpha}$. We call $\boldsymbol{\alpha}$ the **sparse code**.

$$\underbrace{\begin{pmatrix} \mathbf{x} \end{pmatrix}}_{\mathbf{x}\in\mathbb{R}^n} \approx \underbrace{\begin{pmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \cdots & \mathbf{d}_p \end{pmatrix}}_{\mathbf{D}\in\mathbb{R}^{n\times p}} \underbrace{\begin{pmatrix} \boldsymbol{\alpha}[1] \\ \boldsymbol{\alpha}[2] \\ \vdots \\ \boldsymbol{\alpha}[p] \end{pmatrix}}_{\boldsymbol{\alpha}\in\mathbb{R}^p, \textbf{sparse}}$$

The modern parsimony and the ℓ_1-norm

Sparse linear models: machine learning/statistics point of view

Let $(y_i, \mathbf{x}_i)_{i=1}^n$ be a training set, where the vectors $\mathbf{x}_i$ are in $\mathbb{R}^p$ and are called features. The scalars y_i are in

- $\{-1, +1\}$ for **binary** classification problems.
- $\mathbb{R}$ for **regression** problems.

We assume there exists a relation $y \approx \boldsymbol{\beta}^\top \mathbf{x}$, and solve

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \underbrace{\frac{1}{n} \sum_{i=1}^{n} L(y_i, \boldsymbol{\beta}^\top \mathbf{x}_i)}_{\text{empirical risk}} + \underbrace{\lambda \psi(\boldsymbol{\beta})}_{\text{regularization}} .$$

The modern parsimony and the ℓ_1-norm

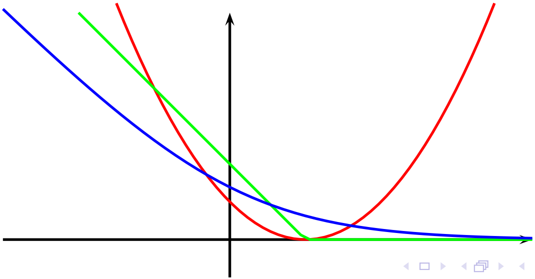
Sparse linear models: machine learning/statistics point of view

A few examples:

Ridge regression:
$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2}(y_i - \boldsymbol{\beta}^\top \mathbf{x}_i)^2 + \lambda \|\boldsymbol{\beta}\|_2^2.$$

Linear SVM:
$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \boldsymbol{\beta}^\top \mathbf{x}_i) + \lambda \|\boldsymbol{\beta}\|_2^2.$$

Logistic regression:
$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \log\left(1 + e^{-y_i \boldsymbol{\beta}^\top \mathbf{x}_i}\right) + \lambda \|\boldsymbol{\beta}\|_2^2.$$



The modern parsimony and the ℓ_1-norm

Sparse linear models: machine learning/statistics point of view

A few examples:

Ridge regression:
$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}(y_i - \boldsymbol{\beta}^\top \mathbf{x}_i)^2 + \lambda\|\boldsymbol{\beta}\|_2^2.$$

Linear SVM:
$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y_i\boldsymbol{\beta}^\top \mathbf{x}_i) + \lambda\|\boldsymbol{\beta}\|_2^2.$$

Logistic regression:
$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n}\log\left(1 + e^{-y_i\boldsymbol{\beta}^\top \mathbf{x}_i}\right) + \lambda\|\boldsymbol{\beta}\|_2^2.$$

The **squared ℓ_2-norm** induces "**smoothness**" in $\boldsymbol{\beta}$. When one knows in advance that $\boldsymbol{\beta}$ should be sparse, one should use a **sparsity-inducing** regularization such as the **ℓ_1-norm**. [Chen et al., 1999, Tibshirani, 1996]

The modern parsimony and the ℓ_1-norm

Originally used to induce sparsity in geophysics [Claerbout and Muir, 1973, Taylor et al., 1979], the ℓ_1-norm became popular in statistics with the **Lasso** [Tibshirani, 1996] and in signal processing with the **Basis pursuit** [Chen et al., 1999].

Three "equivalent" formulations

1. $$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1;$$
2. $$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\|_1 \leq \mu;$$
3. $$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \|\boldsymbol{\alpha}\|_1 \ \text{ s.t. } \ \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \leq \varepsilon.$$

The modern parsimony and the ℓ_1-norm

And some variants...

For noiseless problems

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \|\boldsymbol{\alpha}\|_1 \;\; \text{s.t.} \;\; \mathbf{x} = \mathbf{D}\boldsymbol{\alpha}.$$

Beyond least squares

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} f(\boldsymbol{\alpha}) + \lambda \|\boldsymbol{\alpha}\|_1,$$

where $f : \mathbb{R}^p \to \mathbb{R}$ is convex.

The modern parsimony and the ℓ_1-norm

And some variants...

For noiseless problems

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \|\boldsymbol{\alpha}\|_1 \;\; \text{s.t.} \;\; \mathbf{x} = \mathbf{D}\boldsymbol{\alpha}.$$

Beyond least squares

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} f(\boldsymbol{\alpha}) + \lambda \|\boldsymbol{\alpha}\|_1,$$

where $f : \mathbb{R}^p \to \mathbb{R}$ is convex.

An important question remains:

why does the ℓ_1-norm induce sparsity?

The modern parsimony and the ℓ_1-norm

Why does the ℓ_1-norm induce sparsity?

Can we get some intuition from the simplest isotropic case?

$$\hat{\boldsymbol{\alpha}}(\lambda) = \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1,$$

or equivalently the Euclidean projection onto the ℓ_1-ball?

$$\tilde{\boldsymbol{\alpha}}(\mu) = \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\|_1 \leq \mu.$$

"equivalent" means that for all $\lambda > 0$, there exists $\mu \geq 0$ such that $\tilde{\boldsymbol{\alpha}}(\mu) = \hat{\boldsymbol{\alpha}}(\lambda)$.

The modern parsimony and the ℓ_1-norm

Why does the ℓ_1-norm induce sparsity?

Can we get some intuition from the simplest isotropic case?

$$\hat{\boldsymbol{\alpha}}(\lambda) = \underset{\boldsymbol{\alpha}\in\mathbb{R}^p}{\arg\min} \frac{1}{2}\|\mathbf{x} - \boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1,$$

or equivalently the Euclidean projection onto the ℓ_1-ball?

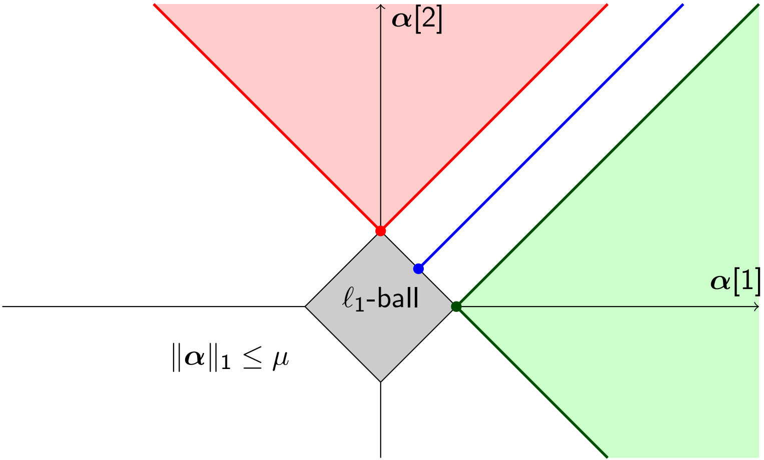
$$\tilde{\boldsymbol{\alpha}}(\mu) = \underset{\boldsymbol{\alpha}\in\mathbb{R}^p}{\arg\min} \frac{1}{2}\|\mathbf{x} - \boldsymbol{\alpha}\|_2^2 \;\text{ s.t. }\; \|\boldsymbol{\alpha}\|_1 \leq \mu.$$

"equivalent" means that for all $\lambda > 0$, there exists $\mu \geq 0$ such that $\tilde{\boldsymbol{\alpha}}(\mu) = \hat{\boldsymbol{\alpha}}(\lambda)$.

The relation between μ and λ is unknown a priori.

Why does the ℓ_1-norm induce sparsity?

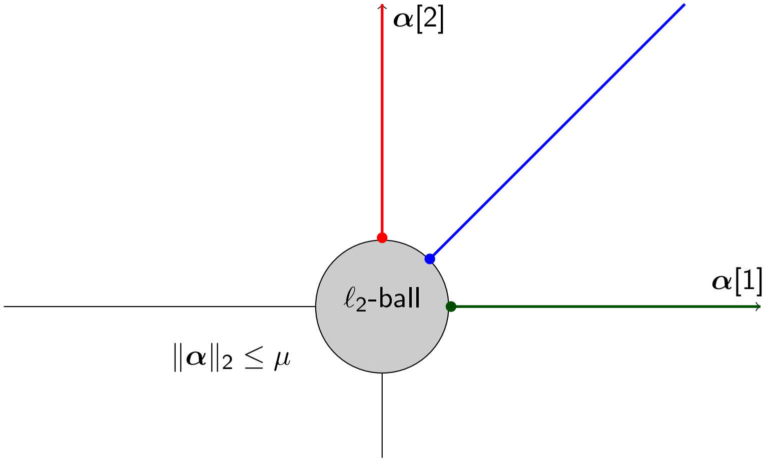
Regularizing with the ℓ_1-norm



The projection onto a convex set is “biased” towards singularities.

Why does the ℓ_1-norm induce sparsity?

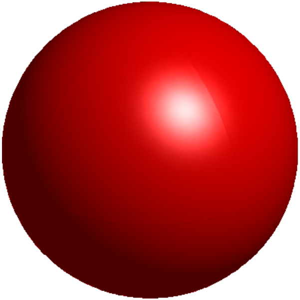
Regularizing with the ℓ_2-norm

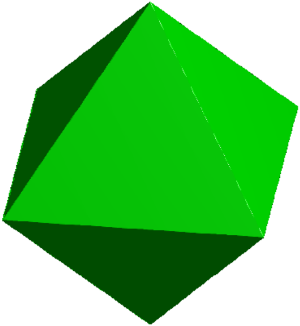


The ℓ_2-norm is isotropic.

Why does the ℓ_1-norm induce sparsity?

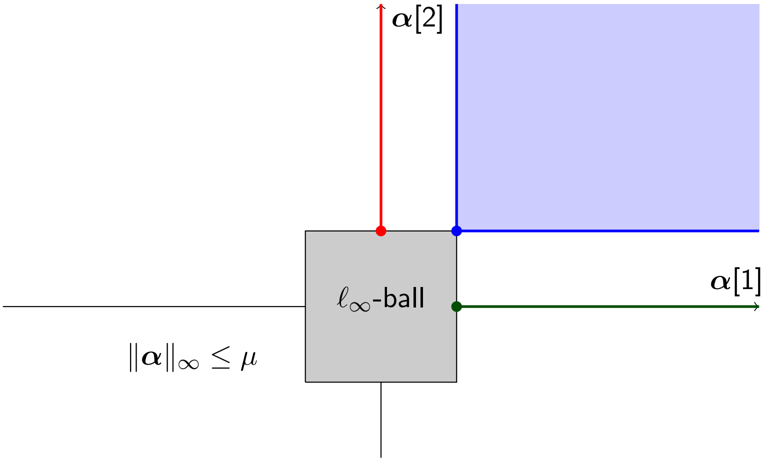
In 3D. (images produced by G. Obozinski)





Why does the ℓ_1-norm induce sparsity?

Regularizing with the ℓ_∞-norm



The ℓ_∞-norm encourages $|\alpha[1]| = |\alpha[2]|$.

Why does the ℓ_1-norm induce sparsity?

Analytical point of view: 1D case

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

Piecewise quadratic function with a kink at zero.

Derivative at 0_+: $g_+ = -x + \lambda$ and 0_-: $g_- = -x - \lambda$.

Optimality conditions. α is optimal iff:

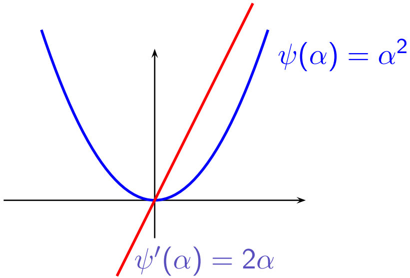
- $|\alpha| > 0$ and $(x - \alpha) + \lambda \, \text{sign}(\alpha) = 0$
- $\alpha = 0$ and $g_+ \geq 0$ and $g_- \leq 0$

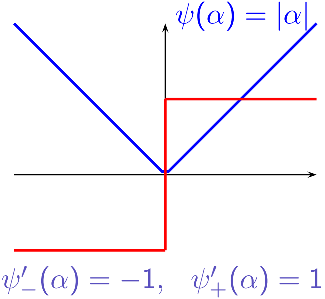
The solution is a **soft-thresholding**:

$$\alpha^\star = \text{sign}(x)(|x| - \lambda)^+.$$

Why does the ℓ_1-norm induce sparsity?

Comparison with ℓ_2-regularization in 1D

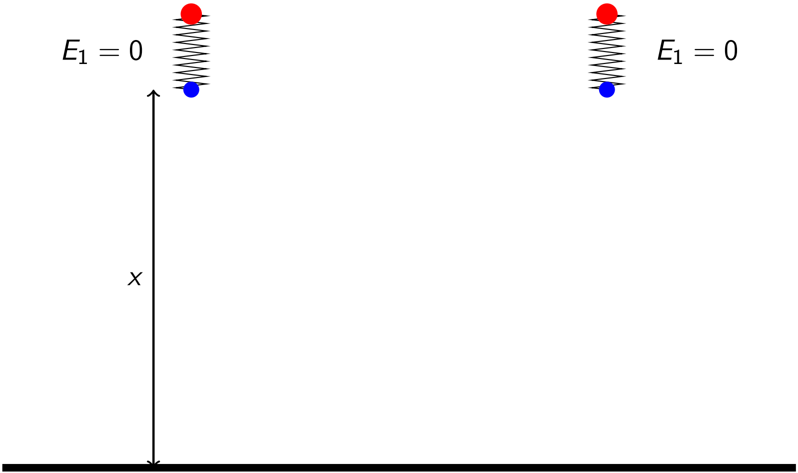




The gradient of the ℓ_2-penalty vanishes when α get close to 0. On its differentiable part, the norm of the gradient of the ℓ_1-norm is constant.

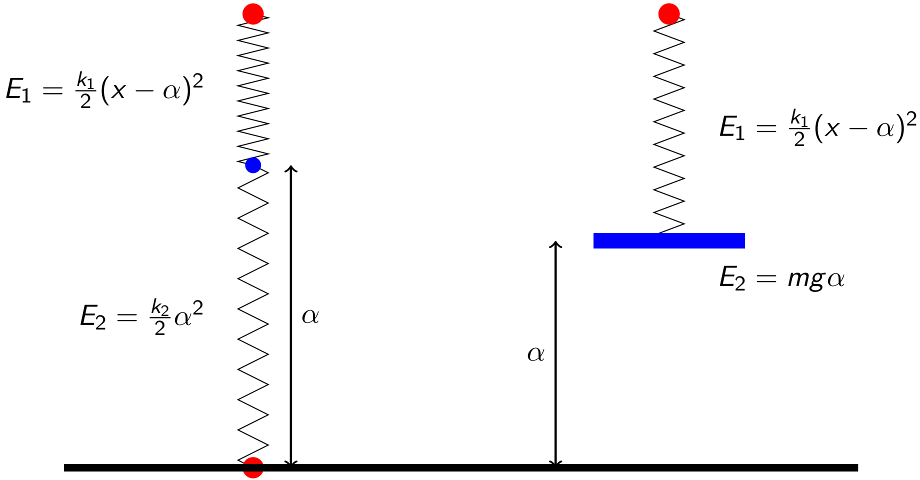
Why does the ℓ_1-norm induce sparsity?

Physical illustration



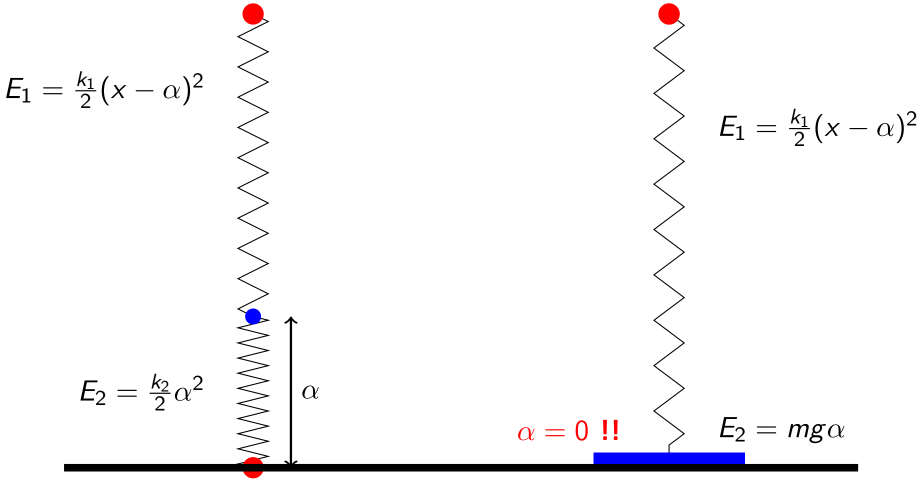
Why does the ℓ_1-norm induce sparsity?

Physical illustration



Why does the ℓ_1-norm induce sparsity?

Physical illustration



Why does the ℓ_1-norm induce sparsity?

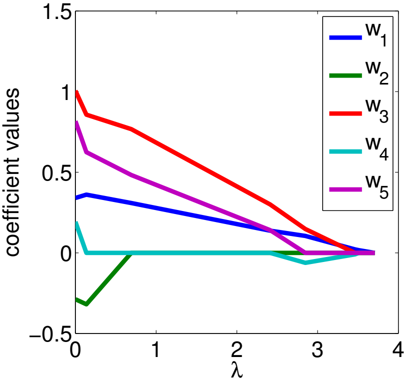


Figure: The regularization path of the Lasso.

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1.$$

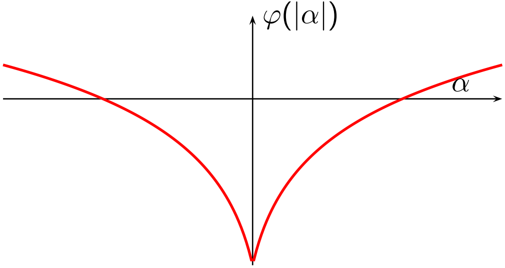
Non-convex sparsity-inducing penalties

Exploiting concave functions with a kink at zero

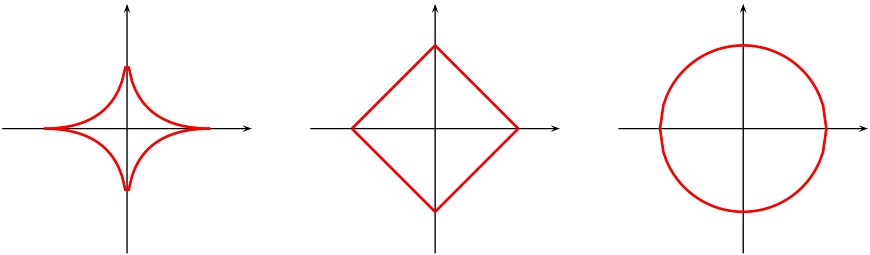
$\psi(\boldsymbol{\alpha}) = \sum_{j=1}^{p} \varphi(|\boldsymbol{\alpha}[j]|)$.

- ℓ_q-penalty, with $0 < q < 1$: $\psi(\boldsymbol{\alpha}) \triangleq \sum_{j=1}^{p} |\boldsymbol{\alpha}[j]|^q$, [Frank and Friedman, 1993];
- log penalty, $\psi(\boldsymbol{\alpha}) \triangleq \sum_{j=1}^{p} \log(|\boldsymbol{\alpha}[j]| + \varepsilon)$.

φ is any function that looks like this:



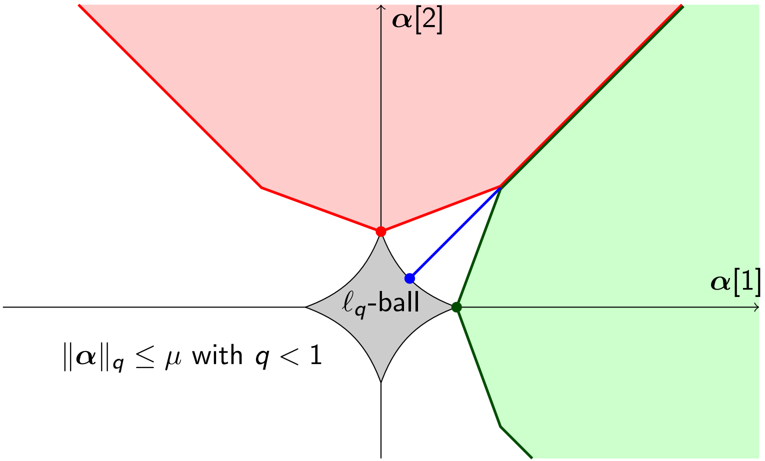
Non-convex sparsity-inducing penalties



(a) $\ell_{0.5}$-ball, 2-D (b) ℓ_1-ball, 2-D (c) ℓ_2-ball, 2-D

Figure: Open balls in 2-D corresponding to several ℓ_q-norms and pseudo-norms.

Non-convex sparsity-inducing penalties

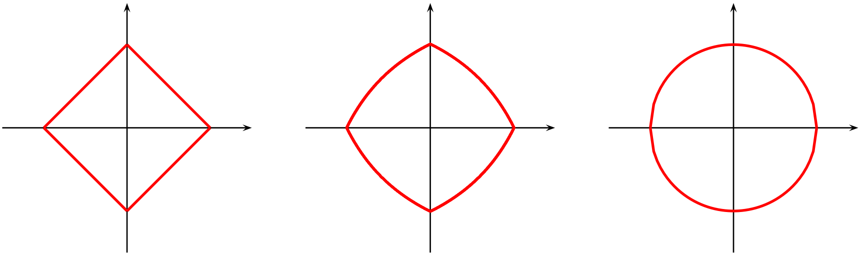


Elastic-net

The **elastic net** introduced by [Zou and Hastie, 2005]

$$\psi(\boldsymbol{\alpha}) = \|\boldsymbol{\alpha}\|_1 + \gamma\|\boldsymbol{\alpha}\|_2^2,$$

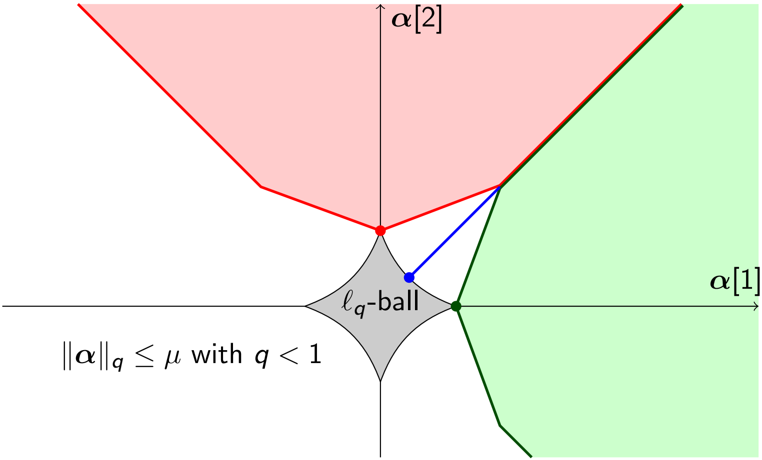
The penalty provides more stable (but less sparse) solutions.



(a) ℓ_1-ball, 2-D (b) elastic-net, 2-D (c) ℓ_2-ball, 2-D

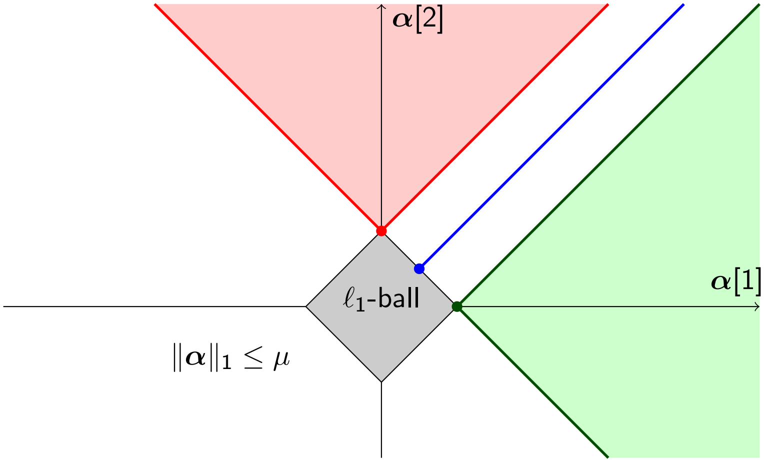
The elastic-net

vs other penalties



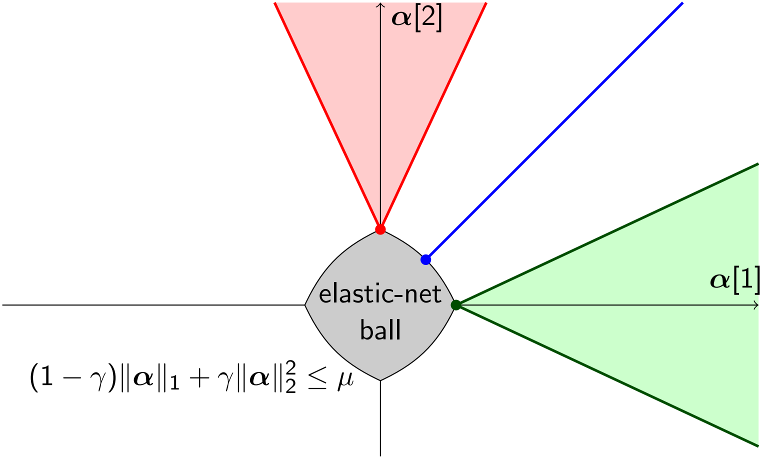
The elastic-net

vs other penalties



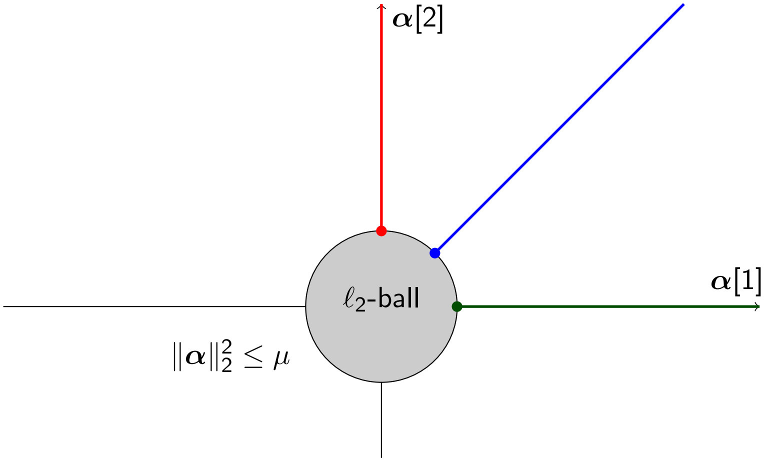
The elastic-net

vs other penalties



The elastic-net

vs other penalties



Total variation and fused Lasso

The anisotropic **total variation** [Rudin et al., 1992]

$$\psi(\boldsymbol{\alpha}) = \sum_{j=1}^{p-1} |\boldsymbol{\alpha}[j+1] - \boldsymbol{\alpha}[j]|,$$

called **fused Lasso** in statistics [Tibshirani et al., 2005]. The penalty encourages piecewise constant signals (can be extended to images).

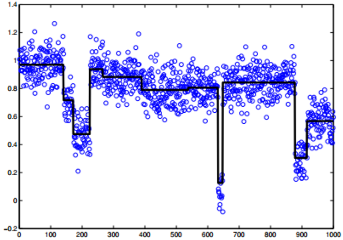


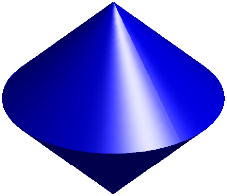
Image borrowed from a talk of J.-P. Vert, representing DNA copy numbers.

Group Lasso and mixed norms

[Turlach et al., 2005, Yuan and Lin, 2006, Zhao et al., 2009]
[Grandvalet and Canu, 1999, Bakin, 1999]

the ℓ_1/ℓ_q-norm : $$\psi(\boldsymbol{\alpha}) = \sum_{g \in \mathcal{G}} \|\boldsymbol{\alpha}[g]\|_q.$$

- $\mathcal{G}$ is a **partition** of $\{1, \ldots, p\}$;
- $q = 2$ or $q = \infty$ in practice;
- can be interpreted as the ℓ_1-norm of $[\|\boldsymbol{\alpha}[g]\|_q]_{g \in \mathcal{G}}$.



$$\psi(\boldsymbol{\alpha}) = \|\boldsymbol{\alpha}[\{1, 2\}]\|_2 + |\boldsymbol{\alpha}[3]|.$$

Spectral sparsity

[Fazel et al., 2001, Srebro et al., 2005]

A natural regularization function for matrices is the **rank**

$$\text{rank}(\mathbf{A}) \triangleq |\{j : s_j(\mathbf{A}) \neq 0\}| = \|\mathbf{s}(\mathbf{A})\|_0,$$

where s_j is the j-th singular value and $\mathbf{s}$ is the *spectrum* of $\mathbf{A}$.

A successful convex relaxation of the rank is the sum of singular values

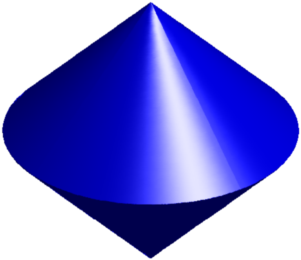
$$\|\mathbf{A}\|_* \triangleq \sum_{j=1}^{p} s_j(\mathbf{A}) = \|\mathbf{s}(\mathbf{A})\|_1,$$

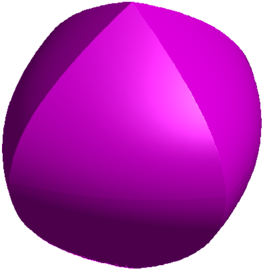
for $\mathbf{A}$ in $\mathbb{R}^{p \times k}$ with $k \geq p$.

The resulting function is a norm, called the **trace** or **nuclear** norm.

Structured sparsity

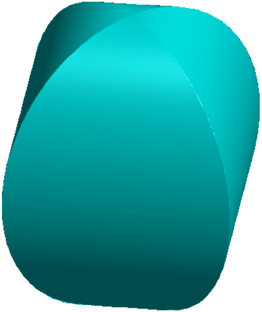
images produced by G. Obozinski

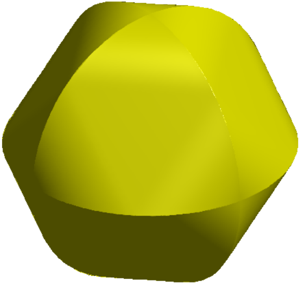




Structured sparsity

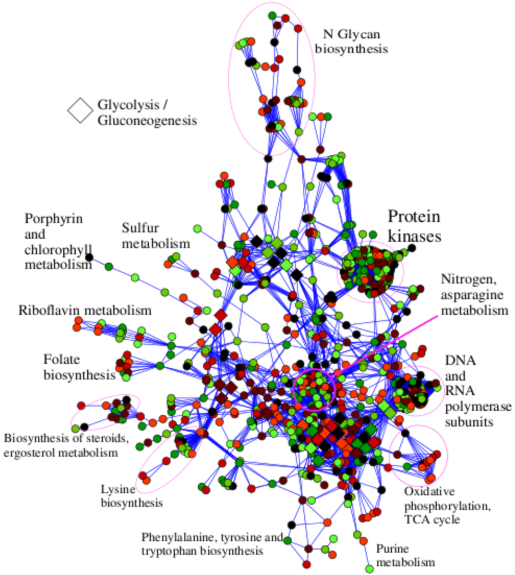
images produced by G. Obozinski





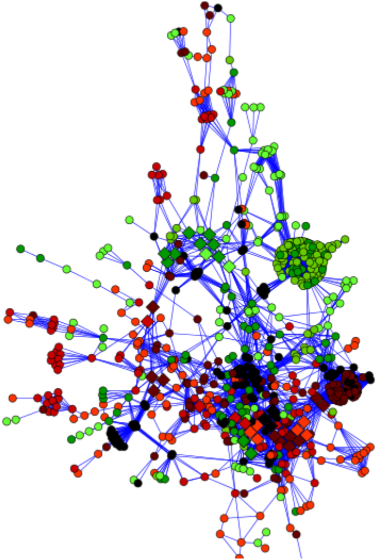
Structured sparsity

Metabolic network of the budding yeast from Rapaport et al. [2007]



Structured sparsity

Metabolic network of the budding yeast from Rapaport et al. [2007]



Structured sparsity

Warning: Under the name "structured sparsity" appear in fact significantly different formulations!

1. non-convex
   - zero-tree wavelets [Shapiro, 1993];
   - predefined collection of sparsity patterns: [Baraniuk et al., 2010];
   - **select a union of groups: [Huang et al., 2009]**;
   - structure via Markov random fields: [Cehver et al., 2008];
2. convex (norms)
   - **tree-structure: [Zhao et al., 2009]**;
   - **select a union of groups: [Jacob et al., 2009]**;
   - **zero-pattern is a union of groups: [Jenatton et al., 2011a]**;
   - other norms: [Micchelli et al., 2013].

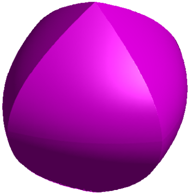
Structured sparsity

Group Lasso with overlapping groups [Jenatton et al., 2011a]

$$\psi(\boldsymbol{\alpha}) = \sum_{g \in \mathcal{G}} \|\boldsymbol{\alpha}[g]\|_q.$$

What happens when the groups overlap?

- the pattern of non-zero variables is an intersection of groups;
- the **zero pattern is a union of groups**.



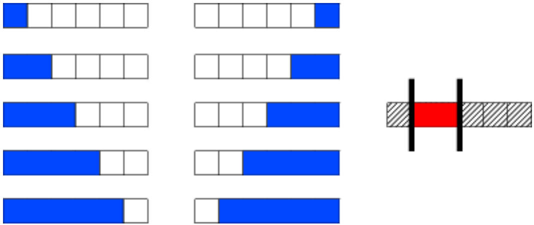
$$\psi(\boldsymbol{\alpha}) = \|\boldsymbol{\alpha}\|_2 + |\boldsymbol{\alpha}[2]| + |\boldsymbol{\alpha}[3]|.$$

Structured sparsity

Group Lasso with overlapping groups [Jenatton et al., 2011a]

Examples of set of groups $\mathcal{G}$

Selection of contiguous patterns on a sequence, $p = 6$.



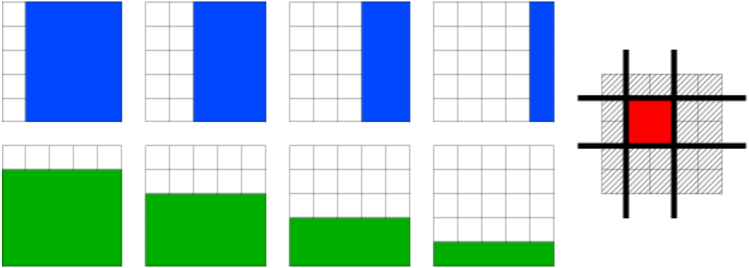
- $\mathcal{G}$ is the set of blue groups.
- Any union of blue groups set to zero leads to the selection of a contiguous pattern.

Structured sparsity

Group Lasso with overlapping groups [Jenatton et al., 2011a]

Examples of set of groups $\mathcal{G}$

Selection of rectangles on a 2-D grids, $p = 25$.



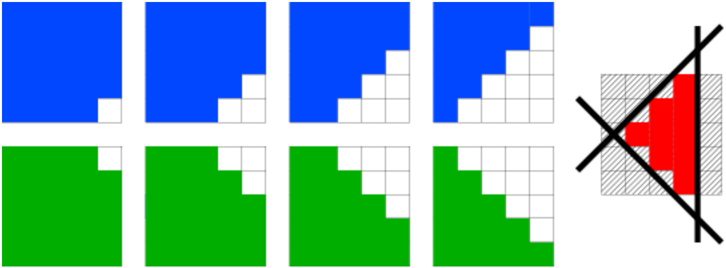
- $\mathcal{G}$ is the set of blue/green groups (with their not displayed complements).

Structured sparsity

Group Lasso with overlapping groups [Jenatton et al., 2011a]

Examples of set of groups $\mathcal{G}$

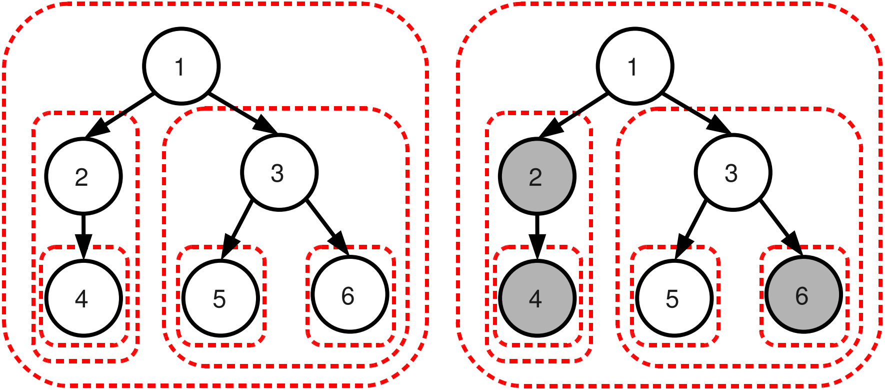
Selection of diamond-shaped patterns on a 2-D grids, $p = 25$.



- It is possible to extent such settings to 3-D space, or more complex topologies.

Structured sparsity

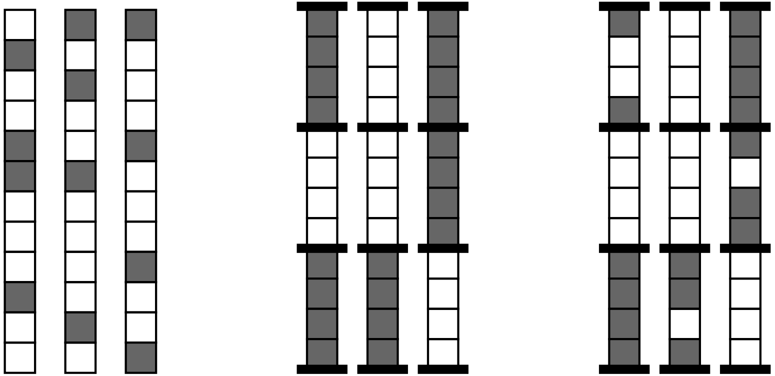
Hierarchical norms [Zhao et al., 2009].



A node can be active only if its **ancestors are active**.
The selected patterns are **rooted subtrees.**

Structured sparsity

Hierarchical norms [Zhao et al., 2009].



(d) Sparsity. (e) Group sparsity. (f) Hierarchical sparsity.

Structured sparsity

the non-convex penalty of Huang et al. [2009]

Warning: different point of view than in the previous three slides

$$\varphi(\boldsymbol{\alpha}) \triangleq \min_{\mathcal{J} \subseteq \mathcal{G}} \left\{ \sum_{g \in \mathcal{J}} \eta_g \ \text{ s.t. } \ \text{Supp}(\boldsymbol{\alpha}) \subseteq \bigcup_{g \in \mathcal{J}} g \right\}.$$

- the penalty is **non-convex**.
- is **NP-hard** to compute (set cover problem).
- The pattern of non-zeroes in $\boldsymbol{\alpha}$ is a **union** of (a few) groups.

It can be rewritten as a boolean linear program:

$$\varphi(\boldsymbol{\alpha}) = \min_{\mathbf{x} \in \{0,1\}^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \ \text{ s.t. } \ \mathcal{N}\mathbf{x} \geq \text{Supp}(\boldsymbol{\alpha}) \right\}.$$

Structured sparsity

convex relaxation and the penalty of Jacob et al. [2009]

The penalty of Huang et al. [2009]:

$$\varphi(\boldsymbol{\alpha}) = \min_{\mathbf{x}\in\{0,1\}^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathcal{N}\mathbf{x} \geq \text{Supp}(\boldsymbol{\alpha}) \right\}.$$

A convex LP-relaxation:

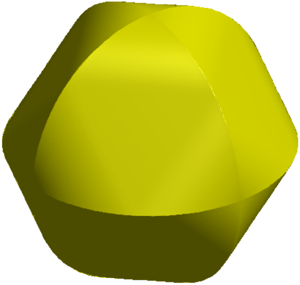
$$\psi(\boldsymbol{\alpha}) \triangleq \min_{\mathbf{x}\in\mathbb{R}_+^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathcal{N}\mathbf{x} \geq |\boldsymbol{\alpha}| \right\}.$$

Lemma: ψ is the penalty of Jacob et al. [2009] with the ℓ_∞-norm:

$$\psi(\boldsymbol{\alpha}) = \min_{(\boldsymbol{\beta}^g\in\mathbb{R}^p)_{g\in\mathcal{G}}} \sum_{g\in\mathcal{G}} \eta_g \|\boldsymbol{\beta}_g\|_\infty \;\; \text{s.t.} \;\; \boldsymbol{\alpha} = \sum_{g\in\mathcal{G}} \boldsymbol{\beta}_g \text{ and } \forall g, \; \text{Supp}(\boldsymbol{\beta}_g) \subseteq g,$$

Structured sparsity

The norm of Jacob et al. [2009] in 3D



$\psi(\boldsymbol{\alpha})$ with $\mathcal{G} = \{\{1,2\},\{2,3\},\{1,3\}\}$.

Sparse recovery: theoretical results for the Lasso

three upcoming slides are inspired from a lecture of G. Obozinski given at Hólar in 2010

Given some observations $(y_i, \mathbf{x}_i)_{i=1,\ldots,n}$, with y_i in $\mathbb{R}$, assume that the linear model $y_i = \mathbf{x}_i^\top \boldsymbol{\theta} + \varepsilon_i$ is valid, with $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$.

Given an estimate $\hat{\boldsymbol{\theta}}$, three main problems:

1. **Regular consistency**: convergence of estimator $\hat{\boldsymbol{\theta}}$ to $\boldsymbol{\theta}$, i.e., $\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_2$ tends to zero when n tends to ∞;
2. **Model selection consistency**: convergence of the sparsity pattern of $\hat{\boldsymbol{\theta}}$ to the pattern of $\boldsymbol{\theta}$;
3. **Efficiency**: convergence of predictions with $\hat{\boldsymbol{\theta}}$ to the predictions with $\boldsymbol{\theta}$, i.e., $\frac{1}{n}\|\mathbf{X}\hat{\boldsymbol{\theta}} - \mathbf{X}\boldsymbol{\theta}\|_2^2$ tends to zero.

Sparse recovery: theoretical results for the Lasso

Conditions on the design for success

- Restricted Isometry Property (RIP)

$$\sqrt{1-\delta_k}\|\boldsymbol{\theta}\| \leq \|\mathbf{X}\boldsymbol{\theta}\| \leq \sqrt{1+\delta_k} \quad \text{for all} \quad \|\boldsymbol{\theta}\|_0 \leq k.$$

  Subsets of size k of the columns of $\mathbf{X}$ should be close to orthogonal.

- Mutual Incoherence Property (MIP)

$$\max_{i \neq j} |\mathbf{x}_i^\top \mathbf{x}_j| < \mu.$$

- Irrepresentable condition (IC)

$$\|\mathbf{Q}_{\mathbf{J}^c\mathbf{J}}\mathbf{Q}_{\mathbf{J}\mathbf{J}}^{-1}\operatorname{sign}(\boldsymbol{\theta}_{\mathbf{J}})\|_\infty \leqslant 1-\gamma \quad \text{with} \quad \mathbf{Q}_{\mathbf{J}\mathbf{J}'} = \mathbf{X}_{\mathbf{J}}^\top \mathbf{X}_{\mathbf{J}'}.$$

- Restricted Eigenvalue condition (RE)

$$\kappa(k)^2 = \min_{|J| \leqslant k} \quad \min_{\Delta,\ \|\Delta_{J^c}\|_1 \leqslant \|\Delta_J\|_1} \frac{\Delta^\top \mathbf{Q} \Delta}{\|\Delta_J\|_2^2} > 0$$

Sparse recovery: theoretical results for the Lasso

Model selection consistency (Lasso)

- Assume $\boldsymbol{\theta}$ sparse and denote $\mathbf{J} = \{j : \boldsymbol{\theta}[j] \neq 0\}$ the nonzero pattern
- **Irrepresentable Condition(γ)** [Zhao and Yu, 2006, Wainwright, 2009]

$$\|\mathbf{Q}_{\mathbf{J}^c\mathbf{J}}\mathbf{Q}_{\mathbf{J}\mathbf{J}}^{-1}\,\text{sign}(\boldsymbol{\theta}_{\mathbf{J}})\|_\infty \leqslant 1-\gamma$$

where $\mathbf{Q} = \lim_{n\to+\infty} \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^\top \in \mathbb{R}^{p\times p}$ (covariance matrix).

- Note that condition depends on $\boldsymbol{\theta}$ and $\mathbf{J}$

Sparse recovery: theoretical results for the Lasso

Model selection consistency (Lasso)

- Assume $\boldsymbol{\theta}$ sparse and denote $\mathbf{J} = \{j : \boldsymbol{\theta}[j] \neq 0\}$ the nonzero pattern
- **Irrepresentable Condition(γ)** [Zhao and Yu, 2006, Wainwright, 2009]

$$\|\mathbf{Q}_{\mathbf{J}^c\mathbf{J}}\mathbf{Q}_{\mathbf{J}\mathbf{J}}^{-1}\operatorname{sign}(\boldsymbol{\theta}_{\mathbf{J}})\|_\infty \leqslant 1 - \gamma$$

where $\mathbf{Q} = \lim_{n\to+\infty} \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^\top \in \mathbb{R}^{p\times p}$ (covariance matrix).

- Note that condition depends on $\boldsymbol{\theta}$ and $\mathbf{J}$

Theorem (Model selection for classical asymptotics (i.e., p fixed))

IC(0) is necessary and IC(γ) for $\gamma > 0$ is sufficient for model selection consistency.

High-dimension ($p \to +\infty$): additional requirements

- Sample size condition : $n > k \log p$
- Requires lower-bound on magnitude of nonzero $\boldsymbol{\theta}[j]$

see Bühlmann and Van De Geer [2011] for a review.

Compressed sensing

Compressed sensing [Candès et al., 2006] says that

- an s-sparse signal $\boldsymbol{\alpha}^\star$ in $\mathbb{R}^p$ can be exactly recovered by observing $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}^\star$ and solving the linear program

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \mathbf{x} = \mathbf{D}\boldsymbol{\alpha},$$

  where $\mathbf{D}$ satisfies the RIP assumption with $\delta_{2s} \leq \sqrt{2} - 1$. Moreover, the convex relaxation is exact.

- matrices $\mathbf{D}$ in $\mathbb{R}^{m \times p}$ satisfying the RIP assumption can be obtained with simple random sampling schemes with

$$m = O(s \log(p/s)).$$

Compressed sensing and sparse recovery

Remarks

- The theory also admits extensions to approximately sparse signals, noisy measurements...
- extensions where $\mathbf{D}$ is replaced by $\mathbf{Z}^\top\mathbf{D}$ where $\mathbf{Z}$ is random and $\mathbf{D}$ deterministic;
- the dictionaries we are using in this lecture do not satisfy RIP;
- sparse estimation and sparse coding is not compressed sensing.

Sparse recovery and compressed sensing

Some thoughts from Hocking [1976]:

> **The problem of selecting a subset of independent or predictor variables is usually described in an idealized setting**. *That is, it is assumed that (a) the analyst has data on a large number of potential variables which include all relevant variables and appropriate functions of them plus, possibly, some other extraneous variables and variable functions and (b) the analyst has available "good" data on which to base the eventual conclusions.* **In practice, the lack of satisfaction of these assumptions may make a detailed subset selection analysis a meaningless exercise.**

Conclusions from the first part

- the sparsity principle has been used for a long time, and this is not a recent idea;
- there are numerous ways of designing sparse regularization functions adapted to a particular problem. Choosing the best one is not easy and requires some domain knowledge;
- the dictionaries we will use in this literature almost never satisfy theoretical assumptions ensuring sparse recovery.

Other take-home messages:

- sparsity is not always good. If possible, try ℓ_2 before trying ℓ_1;
- convexity is not always good. When trying ℓ_1, try also ℓ_0.

Part II: Discovering the structure of natural images

Dictionary learning

The goal of automatically learning local structures in natural images was first achieved by neuroscientists.

The model of Olshausen and Field [1996] looks for a dictionary $\mathbf{D}$ adapted to a training set of natural image patches $\mathbf{x}_i$, $i = 1, \ldots, n$:

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\psi(\boldsymbol{\alpha}_i),$$

where $\mathbf{A} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_n]$ and $\mathcal{C} \triangleq \{\mathbf{D} \in \mathbb{R}^{m\times p} : \forall\, j,\ \|\mathbf{d}_j\|_2 \leq 1\}$.

Typical settings

- $n \approx 100\,000$;
- $m = 10 \times 10$ pixels;
- $p = 256$.

Dictionary learning

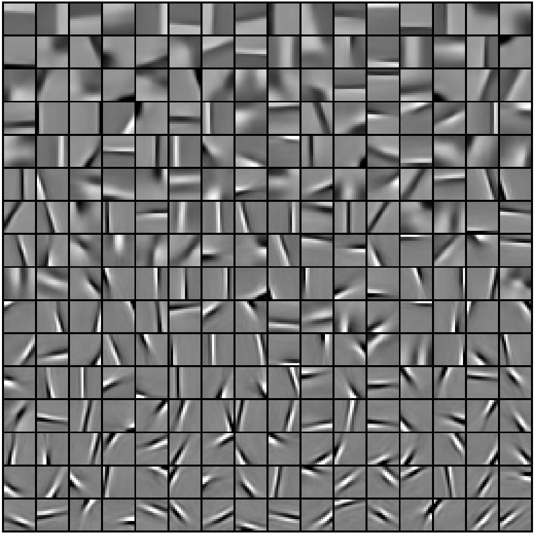


Figure: with centering

Dictionary learning

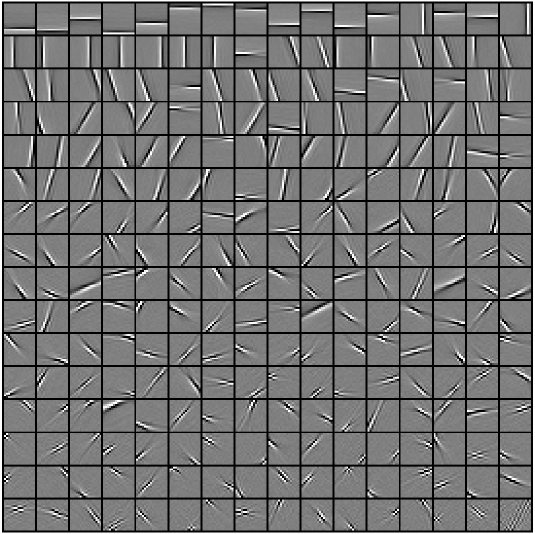


Figure: with whitening

Dictionary learning

Why was it found impressive by neuroscientists?

- since Hubel and Wiesel [1968], it is known that some visual neurons are responding to particular image features, such as oriented edges.
- Later, Daugman [1985] demonstrated that fitting a linear model to neuronal responses given a visual stimuli may produce filters that can be well approximated by a two-dimensional Gabor function.
- the original motivation of Olshausen and Field [1996] was to establish a relation between the statistical structure of natural images and the properties of neurons from area V1.

The results provided some “support” for classical models of V1 based on Gabor filters.

Dictionary learning

Why was it found impressive by neuroscientists?

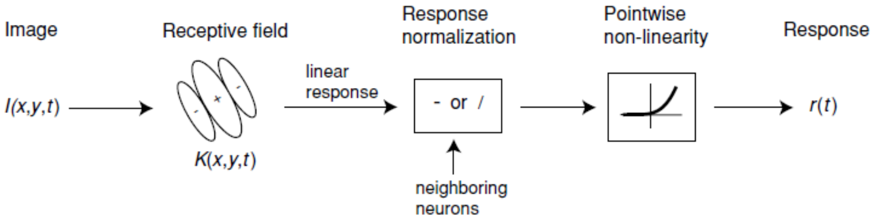
- since Hubel and Wiesel [1968], it is known that some visual neurons are responding to particular image features, such as oriented edges.
- Later, Daugman [1985] demonstrated that fitting a linear model to neuronal responses given a visual stimuli may produce filters that can be well approximated by a two-dimensional Gabor function.
- the original motivation of Olshausen and Field [1996] was to establish a relation between the statistical structure of natural images and the properties of neurons from area V1.

Warning

In fact, little is known about the early visual cortex [Olshausen and Field, 2005, Carandini et al., 2005].

Dictionary learning

Snippet from Olshausen and Field [2005]



However, [...], there remains a great deal that is still unknown about how V1 works and its role in visual system function. **We believe it is quite probable that the correct theory of V1 is still far afield from the currently proposed theories.**

Dictionary learning

Point of views

Matrix factorization

It is useful to see dictionary learning as a matrix factorization problem

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \frac{1}{2n}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_{\mathsf{F}}^2 + \lambda\Psi(\mathbf{A}).$$

This is simply a matter of notation:

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{x}_i-\mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\psi(\boldsymbol{\alpha}_i),$$

but the matrix factorization point of view allows us to make connections with numerous other unsupervised learning techniques, such as K-means, PCA, NMF, ICA...

Dictionary learning

Point of views

Empirical risk minimization

$$\min_{\mathbf{D}\in\mathcal{C}} \frac{1}{n}\sum_{i=1}^{n} L(\mathbf{x}_i, \mathbf{D}),$$

with

$$L(\mathbf{x}, \mathbf{D}) \triangleq \min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\psi(\boldsymbol{\alpha}).$$

Again, this is a matter of notation, but the empirical risk minimization point of view paves the way to

- stochastic optimization [Mairal et al., 2010a];
- some theoretical analysis [**?**Vainsencher et al., 2011, Gribonval et al., 2013].

Dictionary learning

Constrained variants

The formulations below are not equivalent

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \sum_{i=1}^{n} \frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 \ \text{ s.t. } \ \psi(\boldsymbol{\alpha}_i) \leq \mu.$$

or

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \sum_{i=1}^{n} \psi(\boldsymbol{\alpha}_i) \ \text{ s.t. } \ \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 \leq \varepsilon.$$

Using one instead of another is a **matter of taste and of the problem at hand**.

Pre-processing of natural image patches

Centering (also called removing the DC component)

$$\mathbf{x}_i \leftarrow \mathbf{x}_i - \left(\frac{1}{m} \sum_{j=1}^{m} \mathbf{x}_i[j] \right) \mathbf{1}_m,$$

(a) Without pre-processing.

(b) After centering.

Pre-processing of natural image patches

Contrast (variance) normalization

$$\mathbf{x}_i \leftarrow \frac{1}{\max(\|\mathbf{x}_i\|_2, \eta)} \mathbf{x}_i.$$

ex: η can be 0.2 times the mean value of the $\|\mathbf{x}_i\|_2$.

(a) After centering.

(b) After contrast normalization.

Pre-processing of natural image patches

Whitening after centering

$$\mathbf{x}_i \leftarrow \mathbf{U}\mathbf{S}^{\dagger}\mathbf{U}^{\top}\mathbf{x}_i,$$

where $(1/\sqrt{n})\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^{\top}$ (SVD). Sometimes, small singular values are also set to zero. The resulting covariance $(1/n)\mathbf{X}\mathbf{X}^{\top}$ is close to identity.

(a) After centering.

(b) After whitening.

Pre-processing of natural image patches

Treatment of color image patches

Should we use RGB?

Should we use RGB?

- RGB dates back to our first understanding of the nature of light: color spectrum [Newton, 1675], **trichromatic vision** [Young, 1845], color composition [Grassmann, 1854, Maxwell, 1860, von Helmholtz, 1852], biological photoreceptors [Nathans et al., 1986];
- other color spaces, such as CIELab, YIQ, YCrBr have **less correlated** color channels [Pratt, 1971, Sharma and Trussell, 1997], and provide a **better perceptual distance**;
- it does not mean that RGB should never be used: **changing the color space will also change the nature of the noise**...

Pre-processing of natural image patches

Treatment of color image patches

Should we use RGB?

- RGB dates back to our first understanding of the nature of light: color spectrum [Newton, 1675], **trichromatic vision** [Young, 1845], color composition [Grassmann, 1854, Maxwell, 1860, von Helmholtz, 1852], biological photoreceptors [Nathans et al., 1986];
- other color spaces, such as CIELab, YIQ, YCrBr have **less correlated** color channels [Pratt, 1971, Sharma and Trussell, 1997], and provide a **better perceptual distance**;
- it does not mean that RGB should never be used: **changing the color space will also change the nature of the noise**...

how do we perform centering, whitening and normalization?

Pre-processing of natural image patches

Treatment of color image patches

Should we use RGB?

- RGB dates back to our first understanding of the nature of light: color spectrum [Newton, 1675], **trichromatic vision** [Young, 1845], color composition [Grassmann, 1854, Maxwell, 1860, von Helmholtz, 1852], biological photoreceptors [Nathans et al., 1986];
- other color spaces, such as CIELab, YIQ, YCrBr have **less correlated** color channels [Pratt, 1971, Sharma and Trussell, 1997], and provide a **better perceptual distance**;
- it does not mean that RGB should never be used: **changing the color space will also change the nature of the noise**...

how do we perform centering, whitening and normalization?

- **center each R,G,B channel independently.**

Pre-processing of natural image patches

(a) Without pre-processing.

(b) After centering.

(c) Centering and normalization.

(d) After whitening.

Principal component analysis (PCA)

Also known has the Karhunen-Loève or Hotelling transform [Hotelling, 1933], it is often presented as an iterative process finding orthogonal directions maximizing variance in the data.

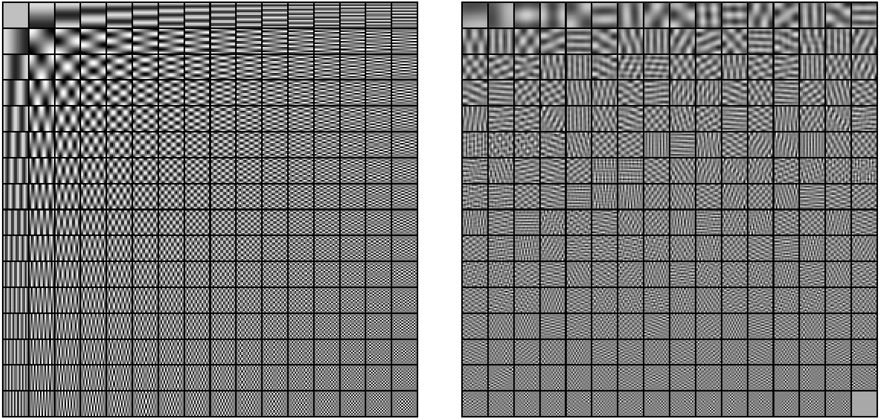
In fact, it can be cast as a **low-rank matrix factorization** problem:

$$\min_{\mathbf{U}\in\mathbb{R}^{m\times k},\mathbf{V}\in\mathbb{R}^{n\times k}} \left\|\mathbf{X} - \mathbf{U}\mathbf{V}^\top\right\|_F^2 \quad \text{s.t.} \quad \mathbf{U}^\top\mathbf{U} = \mathbf{I}_k,$$

where the rows of $\mathbf{X}$ are centered.

As a consequence of the theorem of Eckart and Young [1936], the matrix $\mathbf{U}$ contains the principal components of $\mathbf{X}$ corresponding to the k largest singular values.

Principal component analysis (PCA)

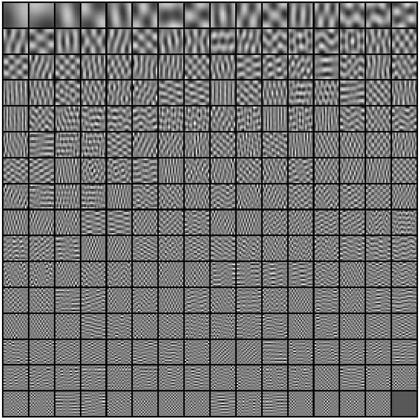


(e) DCT Dictionary. (f) Principal components.

Figure: On the right, we visualize the principal components of 400 000 randomly sampled natural image patches of size 16×16. On the left, we display a discrete cosine transform (DCT) dictionary [Ahmed et al., 1974].

Principal component analysis (PCA)

(a) Original Image.



(b) Principal components.

Figure: Visualization of the principal components of all overlapping patches from the image tiger. Even though the image is not natural, its principal components are similar to the previous ones.

Principal component analysis (PCA)

[Bossomaier and Snyder, 1986, Simoncelli and Olshausen, 2001, Hyvärinen et al., 2009].

Warning

The sinusoids produced by PCA have **nothing to do with the structure of natural images**, but are due to a property of **shift invariance**.

Principal component analysis (PCA)

[Bossomaier and Snyder, 1986, Simoncelli and Olshausen, 2001, Hyvärinen et al., 2009].

Warning

The sinusoids produced by PCA have **nothing to do with the structure of natural images**, but are due to a property of **shift invariance**.

Consider an infinite 1D signal with covariance $\Sigma[k,l] = \sigma(k-l)$, where σ is even. Then, for all ω and φ,

$$\sum_l \Sigma(k,l) e^{i(\omega l + \varphi)} = \sum_l \sigma(l-k) e^{i(\omega l + \varphi)} = \left(\sum_{l'} \sigma(l') e^{i\omega l'} \right) e^{i(\omega k + \varphi)},$$

Since the function σ is even, the infinite sum $\left(\sum_{l'} \sigma(l') e^{i\omega l'} \right)$ is real, and the signals $[\sin(\omega k + \varphi)]_{k \in \mathbb{Z}}$ are all eigenvectors of Σ.

Principal component analysis (PCA)

[Bossomaier and Snyder, 1986, Simoncelli and Olshausen, 2001, Hyvärinen et al., 2009].

Warning

The sinusoids produced by PCA have **nothing to do with the structure of natural images**, but are due to a property of **shift invariance**.

Consider an infinite 1D signal with covariance $\Sigma[k, l] = \sigma(k - l)$, where σ is even. Then, for all ω and φ,

$$\sum_{l} \Sigma(k, l) e^{i(\omega l + \varphi)} = \sum_{l} \sigma(l - k) e^{i(\omega l + \varphi)} = \left(\sum_{l'} \sigma(l') e^{i\omega l'} \right) e^{i(\omega k + \varphi)},$$

Since the function σ is even, the infinite sum $\left(\sum_{l'} \sigma(l') e^{i\omega l'}\right)$ is real, and the signals $[\sin(\omega k + \varphi)]_{k \in \mathbb{Z}}$ are all eigenvectors of Σ.

Note that controlling the approximation of the PCs by the discrete Fourier transform for finite signals is non-trivial [Pearl, 1973].

Clustering or vector quantization

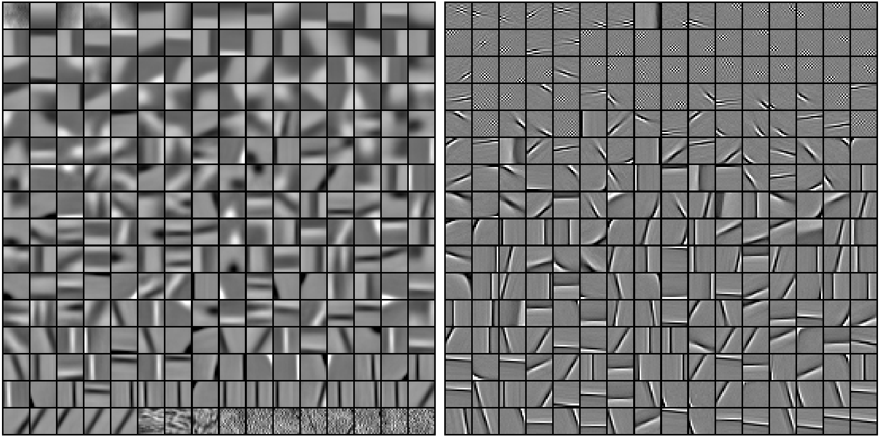
First used on natural image patches for **compression and communication** purposes [Nasrabadi and King, 1988, Gersho and Gray, 1992]. The goal is to find p clusters in the data, by minimizing the following objective:

$$\min_{\substack{\mathbf{D} \in \mathbb{R}^{m \times p} \\ \forall i,\ l_i \in \{1,\dots,p\}}} \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{d}_{l_i}\|_2^2, \qquad (1)$$

This is again a matrix factorization problem

$$\min_{\substack{\mathbf{D} \in \mathbb{R}^{m \times p} \\ \mathbf{A} \in \{0,1\}^{p \times n}}} \frac{1}{2n}\|\mathbf{X} - \mathbf{DA}\|_F^2 \ \text{ s.t. } \ \forall i,\ \sum_{j=1}^{p} \alpha_i[j] = 1.$$

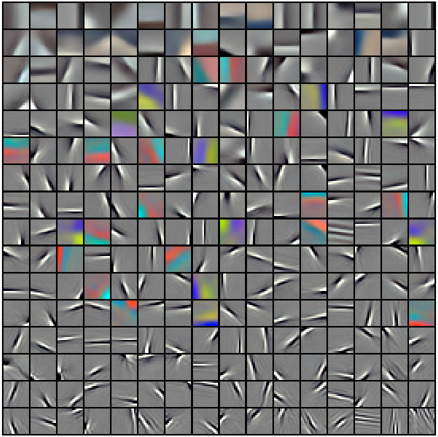
Clustering or vector quantization



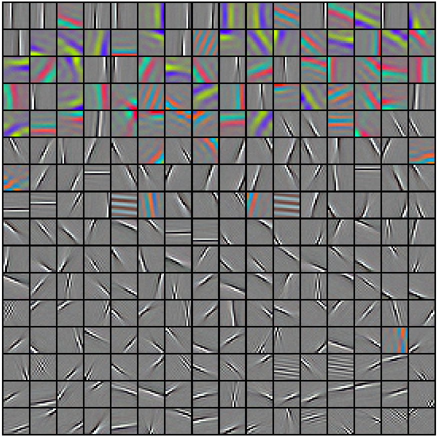
(a) With centering. (b) With whitening.

Figure: Visualization of $p = 256$ centroids computed with the algorithm K-means on $n = 400\,000$ image patches of size $m = 16 \times 16$ pixels.

Dictionary learning on color image patches



(a) With centering - RGB.



(b) With whitening - RGB.

Figure: Dictionaries learned on RGB patches.

Dictionary learning with structured sparsity

Formulation

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \frac{1}{2n}\|\mathbf{X}-\mathbf{DA}\|_{\mathsf{F}}^2 + \frac{\lambda}{n}\sum_{i=1}^{n}\sum_{g\in\mathcal{G}}\|\boldsymbol{\alpha}_i[g]\|_q.$$

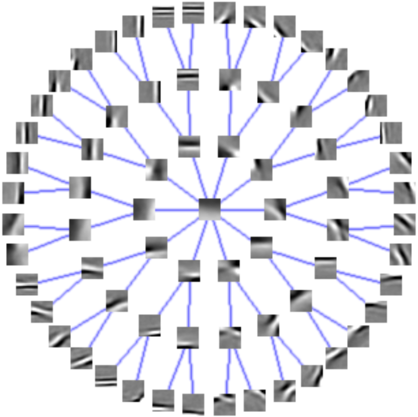
Group structures

- **hierarchical:** organize the dictionary elements in a tree [Jenatton et al., 2010, 2011b];
- **topographic:** organize the elements on a 2D grid [Kavukcuoglu et al., 2009, Mairal et al., 2011]. The groups are 3×3 or 4×4 spatial neighborhoods.

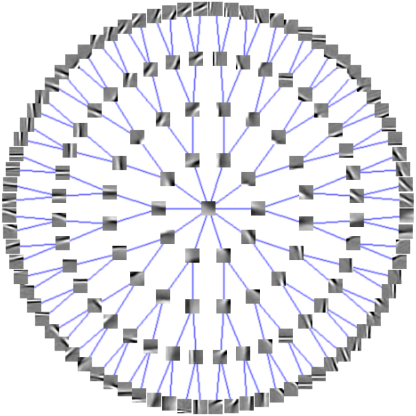
The second group structure is inspired by **topographic ICA** [Hyvärinen et al., 2001].

Dictionary learning with structured sparsity

Hierarchical dictionary learning



(a) Tree structure 1.

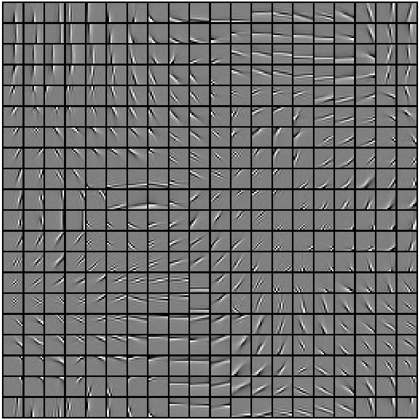


(b) Tree structure 2.

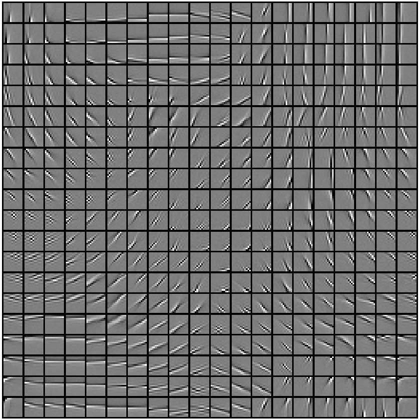
Figure: Hierarchical dictionaries learned on natural image patches of size 16×16 pixels.

Dictionary learning with structured sparsity

Topographic dictionary learning



(a) With 3×3 neighborhoods.



(b) With 4×4 neighborhood.

Figure: Topographic dictionaries learned on whitened natural image patches of size 12×12 pixels.

Other matrix factorization methods

Independent component analysis (ICA)

Assume that $\mathbf{x}$ is a random variable—here a natural image patch— and the columns of $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ are random realizations of $\mathbf{x}$.

ICA is **principle** looking for a factorization $\mathbf{x} = \mathbf{D}\alpha$, where $\mathbf{D}$ is orthogonal and α is a random vector **whose entries are statistically independent** [Bell and Sejnowski, 1997, Hyvärinen et al., 2009].

Other matrix factorization methods

Independent component analysis (ICA)

Assume that $\mathbf{x}$ is a random variable—here a natural image patch— and the columns of $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ are random realizations of $\mathbf{x}$.

ICA is **principle** looking for a factorization $\mathbf{x} = \mathbf{D}\alpha$, where $\mathbf{D}$ is orthogonal and α is a random vector **whose entries are statistically independent** [Bell and Sejnowski, 1997, Hyvärinen et al., 2009].

Warning

Because ICA is only a principle, there is not a unique ICA formulation.

Other matrix factorization methods

Independent component analysis (ICA)

Assume that $\mathbf{x}$ is a random variable—here a natural image patch— and the columns of $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ are random realizations of $\mathbf{x}$.

ICA is **principle** looking for a factorization $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}$, where $\mathbf{D}$ is orthogonal and $\boldsymbol{\alpha}$ is a random vector **whose entries are statistically independent** [Bell and Sejnowski, 1997, Hyvärinen et al., 2009].

How do we measure independence?

Compare $p(\boldsymbol{\alpha})$ with the product of its marginals $\prod_{j=1}^{p} p(\boldsymbol{\alpha}[j])$:

$$KL\left(p(\boldsymbol{\alpha}), \prod_{j=1}^{p} p(\boldsymbol{\alpha}[j])\right) \triangleq \int_{\mathbb{R}^p} p(\boldsymbol{\alpha}) \log\left(\frac{p(\boldsymbol{\alpha})}{\prod_{j=1}^{p} p(\boldsymbol{\alpha}[j])}\right) d\boldsymbol{\alpha},$$

which is zero iff the $\boldsymbol{\alpha}[j]'s$ are independent [Cover and Thomas, 2006].

Other matrix factorization methods

Independent component analysis (ICA)

We can rewrite the Kullback-Leibler distance with entropies

$$KL\left(p(\alpha), \prod_{j=1}^{p} p(\alpha[j])\right) = \sum_{j=1}^{p} H(\alpha[j]) - H(\alpha).$$

The entropy $H(\alpha)$ can be shown to be independent of $\mathbf{D}$ when $\mathbf{D}$ is orthogonal and $\mathbf{x}$ is whitened. Minimizing KL amounts to minimizing

$$\sum_{j=1}^{p} H(\alpha[j]) = \sum_{j=1}^{p} H(\mathbf{d}_j^\top \mathbf{x}).$$

We are close to an ICA "formulation" but not yet there.

The entropy is an abstract quantity that is not computable.

Other matrix factorization methods

Independent component analysis (ICA)

Strategies leading to concrete formulations/algorithms for solving the following problem after whitening the data

$$\min_{\mathbf{D}} \sum_{j=1}^{p} H(\mathbf{d}_j^\top \mathbf{x}) \;\; \text{s.t.} \;\; \mathbf{D}^\top \mathbf{D} = \mathbf{I}.$$

- **parameterizing the densities** $p(\mathbf{d}_j^\top \mathbf{x})$, leading to maximum likelihood estimation [see Hyvärinen et al., 2004];
- plug in **non-parametric estimators** of the entropy [Pham, 2004];
- encourage the distributions of the $\alpha[j]$'s to be **"non-Gaussian"** [Cardoso, 2003].

Among all probability distributions with same variance, the Gaussian ones are known to maximize entropy [Cover and Thomas, 2006].

Other matrix factorization methods

Non-negative matrix factorization [Paatero and Tapper, 1994].

$$\min_{\mathbf{D}\in\mathbb{R}^{m\times p},\mathbf{A}\in\mathbb{R}^{p\times n}} \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_{\mathsf{F}}^2 \;\; \text{s.t.} \;\; \mathbf{D}\geq 0 \;\; \text{and} \;\; \mathbf{A}\geq 0.$$

Other matrix factorization methods

Non-negative matrix factorization [Paatero and Tapper, 1994].

$$\min_{\mathbf{D}\in\mathbb{R}^{m\times p},\mathbf{A}\in\mathbb{R}^{p\times n}} \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 \;\text{ s.t. }\; \mathbf{D}\geq 0 \;\text{ and }\; \mathbf{A}\geq 0.$$

Archetypal analysis [Cutler and Breiman, 1994].

- for all dictionary element j, $\mathbf{d}_j = \mathbf{X}\boldsymbol{\beta}_j$, where $\boldsymbol{\beta}_j$ is in the simplex $\Delta_n \triangleq \{\boldsymbol{\beta}\in\mathbb{R}^n \;\text{ s.t. }\; \boldsymbol{\beta}\geq 0 \;\text{ and }\; \sum_{i=1}^n \boldsymbol{\beta}[i] = 1\}$.
- for all data point i, $\mathbf{x}_i$ is close to $\mathbf{D}\boldsymbol{\alpha}_i$, where $\boldsymbol{\alpha}_i$ is in Δ_p;
- formulation:

$$\min_{\substack{\boldsymbol{\alpha}_i\in\Delta_p \text{ for } 1\leq i\leq n \\ \boldsymbol{\beta}_j\in\Delta_n \text{ for } 1\leq j\leq p}} \|\mathbf{X}-\mathbf{X}\mathbf{B}\mathbf{A}\|_F^2,$$

where $\mathbf{A} = [\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_n]$, $\mathbf{B} = [\boldsymbol{\beta}_1,\ldots,\boldsymbol{\beta}_p]$ and the matrix of archetypes $\mathbf{D}$ is equal to the product $\mathbf{XB}$.

Other matrix factorization methods

Convolutional sparse coding [Zhu et al., 2005, Zeiler et al., 2010]

Main idea

Decompose directly the **full image** $\mathbf{x}$ using **small** dictionary elements placed at all possible positions in the image.

Given a dictionary $\mathbf{D}^{m\times p}$ where m is a patch size, and an image $\mathbf{x}$ in $\mathbb{R}^l$, the image decomposition can be written.

$$\min_{\mathbf{A}\in\mathbb{R}^{p\times l}} \frac{1}{2}\left\|\mathbf{x} - \sum_{k=1}^{l}\mathbf{R}_k^\top \mathbf{D}\boldsymbol{\alpha}_k\right\|_2^2 + \lambda\sum_{i=1}^{l}\|\boldsymbol{\alpha}_k\|_1,$$

Model with effective applications to visual recognition [Zeiler et al., 2010, Rigamonti et al., 2013, Kavukcuoglu et al., 2010].

Then, the extension to dictionary learning is easy.

Other matrix factorization methods

Convolutional sparse coding [Zhu et al., 2005, Zeiler et al., 2010]

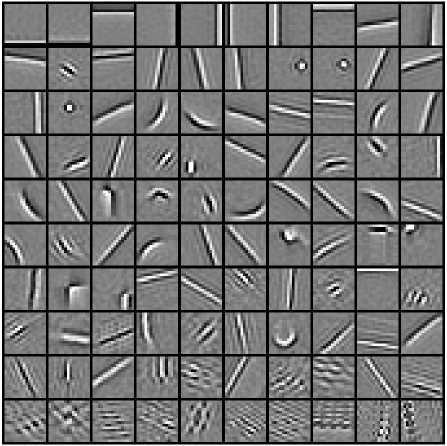


Figure: Visualization of $p = 100$ dictionary elements learned on 30 whitened natural images.

Conclusions from the second part

- intriguing structures naturally emerge from natural images;
- matrix factorization is an effective tool to find these structures;

Advertisement

- some matlab code will be provided upon publication of the monograph for generating most of the figures from this lecture.
- the SPAMS toolbox already contains lots of code (C++ interfaced with Matlab, Python, R) for learning dictionaries, factorizing matrices (NMF, archetypal analysis), solving sparse estimation problems. `http://spams-devel.gforge.inria.fr/`.

Conclusions from the second part

- intriguing structures naturally emerge from natural images;
- matrix factorization is an effective tool to find these structures;

Advertisement

- some matlab code will be provided upon publication of the monograph for generating most of the figures from this lecture.
- the SPAMS toolbox already contains lots of code (C++ interfaced with Matlab, Python, R) for learning dictionaries, factorizing matrices (NMF, archetypal analysis), solving sparse estimation problems. `http://spams-devel.gforge.inria.fr/`.

Question

Is unsupervised learning on natural image patches useful for any prediction task?

Part III: Sparse models for image processing

Image denoising

$$\underbrace{\mathbf{y}}_{\text{measurements}} = \underbrace{\mathbf{x}_{orig}}_{\text{original image}} + \underbrace{\mathbf{w}}_{\text{noise}}$$

Image denoising

Classical image models

$$\underbrace{\mathbf{y}}_{\text{measurements}} = \underbrace{\mathbf{x}_{orig}}_{\text{original image}} + \underbrace{\mathbf{w}}_{\text{noise}}.$$

Energy minimization problem - MAP estimation

$$E(\mathbf{x}) = \underbrace{\frac{1}{2}\|\mathbf{y} - \mathbf{x}\|_2^2}_{\text{relation to measurements}} + \underbrace{\psi(\mathbf{x})}_{\text{image model}}.$$

Some classical priors

- Smoothness $\lambda\|\mathcal{L}\mathbf{x}\|_2^2$;
- total variation $\lambda\|\nabla\mathbf{x}\|_1^2$ [Rudin et al., 1992];
- Markov random fields [Zhu and Mumford, 1997];
- wavelet sparsity $\lambda\|\mathbf{W}\mathbf{x}\|_1$.

Image denoising

The method of Elad and Aharon [2006]

Given a **fixed** dictionary $\mathbf{D}$, a patch $\mathbf{y}_i$ is denoised as follows:

1. center $\mathbf{y}_i$,

$$\mathbf{y}_i^c \triangleq \mathbf{y}_i - \mu_i \mathbf{1}_m \quad \text{with} \quad \mu_i \triangleq \frac{1}{n}\mathbf{1}_m^\top \mathbf{y}_i;$$

2. find a sparse linear combination of dictionary elements that approximates $\mathbf{y}_i^c$ up to the noise level:

$$\min_{\boldsymbol{\alpha}_i \in \mathbb{R}^p} \|\boldsymbol{\alpha}_i\|_0 \ \text{ s.t. } \ \|\mathbf{y}_i^c - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 \leq \varepsilon, \qquad (2)$$

where ε is proportional to the noise variance σ^2;

3. add back the mean component to obtain the clean estimate $\hat{\mathbf{x}}_i$:

$$\hat{\mathbf{x}}_i \triangleq \mathbf{D}\boldsymbol{\alpha}_i^\star + \mu_i \mathbf{1}_m,$$

Image denoising

The method of Elad and Aharon [2006]

An **adaptive** approach

1. extract all overlapping $\sqrt{m} \times \sqrt{m}$ patches $\mathbf{y}_i$.
2. **dictionary learning**: learn $\mathbf{D}$ on the set of centered noisy patches $[\mathbf{y}_1^c, \ldots, \mathbf{y}_n^c]$.
3. **final reconstruction**: find an estimate $\hat{\mathbf{x}}_i$ for every patch using the approach of the previous slide;
4. **patch averaging**:

$$\hat{\mathbf{x}} = \frac{1}{m} \sum_{i=1}^{n} \mathbf{R}_i^\top \hat{\mathbf{x}}_i,$$

Remark

Like other state-of-the-art denoising approaches, it is patch-based [Buades et al., 2005, Dabov et al., 2007].

Practical tricks

- use larger patches when the noise level is high;
- choose $\varepsilon = m(1.15\sigma)^2$ or take the 0.9-quantile of the χ^2_m-distribution.
- always use the ℓ_0 regularization for the final reconstruction;
- using ℓ_1 for learning the dictionary seems to yield better results.

Image inpainting

[Mairal et al., 2008a,b]

For removing small holes in the image, a natural extension consists in introducing a **binary mask** $\mathbf{M}_i$ in the formulation:

$$\min_{\mathbf{D}\in\mathcal{C},\mathbf{A}\in\mathbb{R}^{p\times n}} \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\|\mathbf{M}_i(\mathbf{y}_i-\mathbf{D}\boldsymbol{\alpha}_i)\|_2^2+\lambda\psi(\boldsymbol{\alpha}_i),$$

The approach assumes that

- the noise is not structured;
- the holes are smaller than the patch size.

The problem is called inpainting [Bertalmio et al., 2000].

Image inpainting

[Mairal et al., 2008a,b]

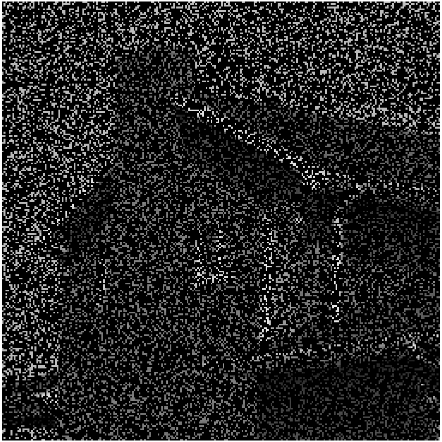


Image inpainting
[Mairal et al., 2008a,b]

Image inpainting

[Mairal et al., 2008a,b]

Image inpainting

Inpainting a 12-Mpixel photograph [Mairal et al., 2009a]

Image inpainting

Inpainting a 12-Mpixel photograph [Mairal et al., 2009a]

Image inpainting

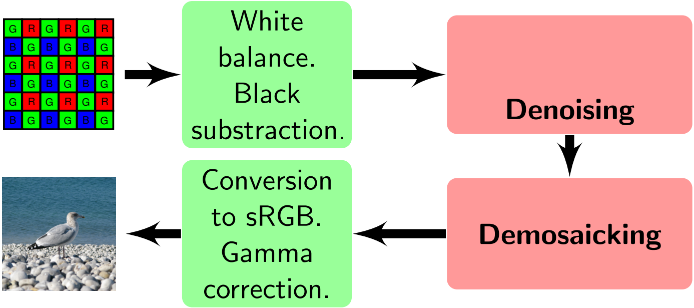
Inpainting a 12-Mpixel photograph [Mairal et al., 2009a]

Image inpainting

Inpainting a 12-Mpixel photograph [Mairal et al., 2009a]

Image demoisaicking

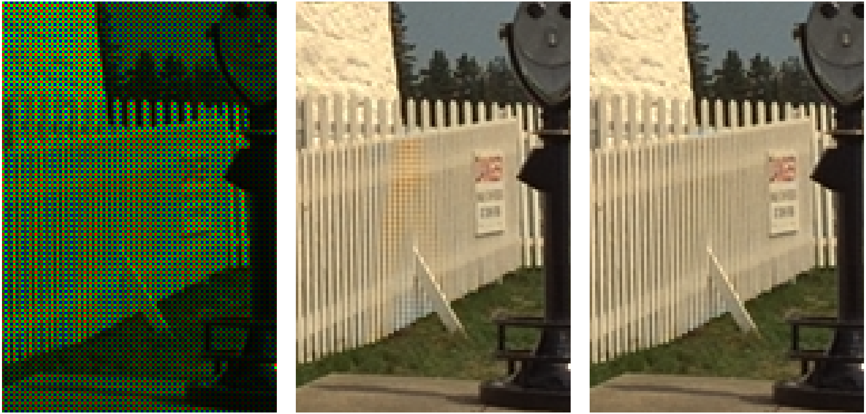
RAW Image Processing



Problem

The noise pattern is very structured: the previous inpainting scheme needs to be modified [Mairal et al., 2008a].

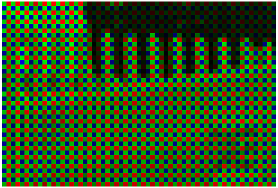
Image demoisaicking



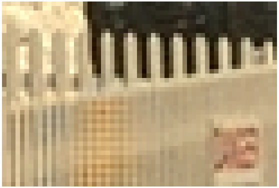
(a) Mosaicked image (b) Demosaicked image A (c) Demosaicked image B

Figure: Demosaicked image A is with the approach previously described; image B is with an extension called non-local sparse model [Mairal et al., 2009b].

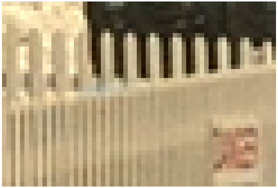
Image demoisaicking



(a) Zoom



(b) Zoom



(c) Zoom

Figure: Demosaicked image A is with the approach previously described; image B is with an extension called non-local sparse model [Mairal et al., 2009b].

Video processing

Extension developed by Protter and Elad [2009]:

Key ideas for video processing

- Using a 3D dictionary.
- Processing of many frames at the same time.
- Dictionary propagation.

Video processing

Inpainting, [Mairal et al., 2008b]

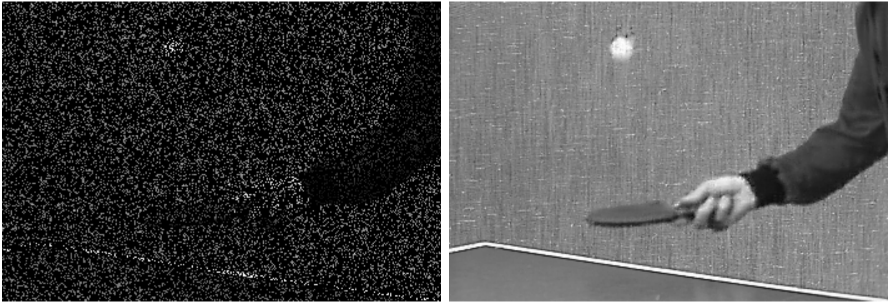


Figure: Inpainting results.

Video processing

Inpainting, [Mairal et al., 2008b]

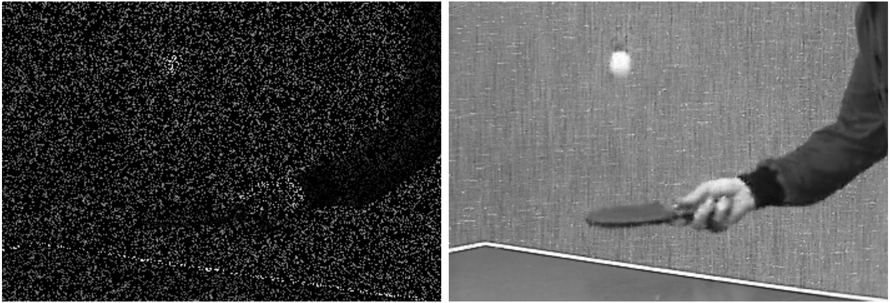


Figure: Inpainting results.

Video processing

Inpainting, [Mairal et al., 2008b]

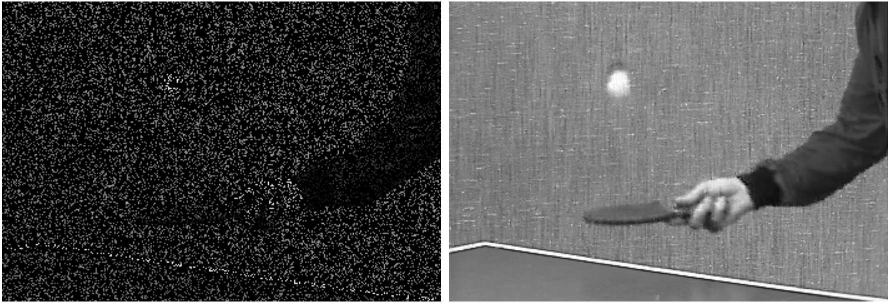


Figure: Inpainting results.

Video processing

Inpainting, [Mairal et al., 2008b]

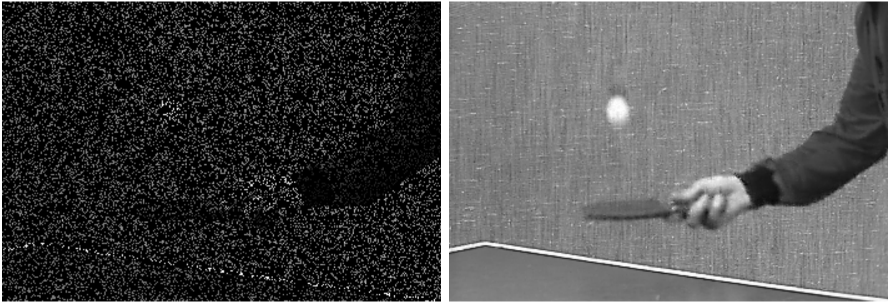
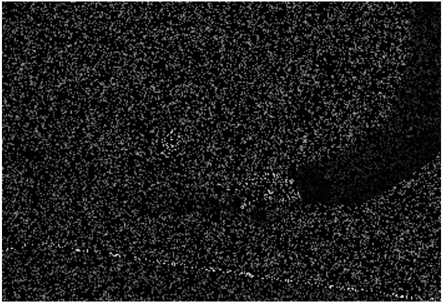


Figure: Inpainting results.

Video processing

Inpainting, [Mairal et al., 2008b]



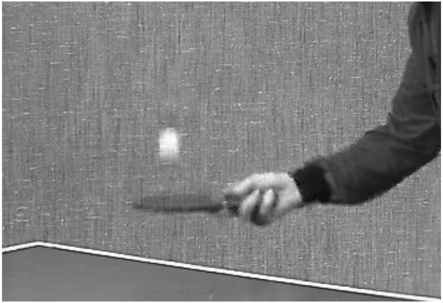


Figure: Inpainting results.

Video processing

Color video denoising, [Mairal et al., 2008b]

Figure: Inpainting results.

Video processing

Color video denoising, [Mairal et al., 2008b]

Figure: Inpainting results.

Video processing

Color video denoising, [Mairal et al., 2008b]

Figure: Inpainting results.

Video processing

Color video denoising, [Mairal et al., 2008b]

Figure: Inpainting results.

Video processing

Color video denoising, [Mairal et al., 2008b]

Figure: Inpainting results.

Image up-scaling

The main recipe of Yang et al. [2010]

The approach requires a database of pairs of training patches $(\mathbf{x}_i^l, \mathbf{x}_i^h)_{i=1}^n$, where $\mathbf{x}_i^l$ in $\mathbb{R}^{m_l}$ is a low-resolution version of the patch $\mathbf{x}_i^h$ in $\mathbb{R}^{m_h}$.

Training step:

$$\min_{\substack{\mathbf{D}_l \in \mathcal{C}_l \\ \mathbf{D}_h \in \mathcal{C}_h \\ \mathbf{A} \in \mathbb{R}^{p \times n}}} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2m_l} \left\| \mathbf{x}_i^l - \mathbf{D}_l \boldsymbol{\alpha}_i \right\|_2^2 + \frac{1}{2m_h} \left\| \mathbf{x}_i^h - \mathbf{D}_h \boldsymbol{\alpha}_i \right\|_2^2 + \lambda \left\| \boldsymbol{\alpha}_i \right\|_1 ,$$

$\mathbf{D}_l$ and $\mathbf{D}_h$ are jointly learned such that the pairs $(\mathbf{x}_i^l, \mathbf{x}_i^h)$ "share" the same sparse decompositions on the dictionaries.

Reconstruction step given a low-resolution image:

$$\min_{\boldsymbol{\beta}_i \in \mathbb{R}^p} \frac{1}{2m_l} \left\| \mathbf{y}_i^l - \mathbf{D}_l \boldsymbol{\beta}_i \right\|_2^2 + \frac{1}{2m_h} \left\| \mathbf{z}_i - \mathbf{D}_h \boldsymbol{\beta}_i \right\|_2^2 + \lambda \left\| \boldsymbol{\beta}_i \right\|_1 ,$$

Image up-scaling

Variant with regression [Zeyde et al., 2012, Couzinie-Devy et al., 2011, Yang et al., 2012]

1. compute $\mathbf{D}_l$ and $\mathbf{A}$ with a classical dictionary learning formulation,

$$\min_{\mathbf{D}_l \in \mathcal{C}_l, \mathbf{A} \in \mathbb{R}^{p \times n}} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} \|\mathbf{x}_i^l - \mathbf{D}_l \boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1.$$

2. obtain $\mathbf{D}_h$ by solving a **multivariate regression** problem:

$$\min_{\mathbf{D}_h \in \mathbb{R}^{m_h \times p}} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} \|\mathbf{x}_i^h - \mathbf{D}_h \boldsymbol{\alpha}_i\|_2^2,$$

where the α_i's are fixed after the first step. See also Zeyde et al. [2012] for other variants.

Main difference with Yang et al. [2010]

- ☺ testing and training is more consistent;
- ☹ $\mathbf{D}_h$ and $\mathbf{D}_l$ are not learned jointly anymore.

Image up-scaling

Variant with task-driven dictionary learning [Couzinie-Devy et al., 2011, Yang et al., 2012]

Define

$$\boldsymbol{\alpha}^\star(\mathbf{x}, \mathbf{D}) \triangleq \underset{\boldsymbol{\alpha} \in \mathbb{R}^p}{\arg\min} \left[\frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \right],$$

Then, the joint dictionary learning formulation consists of minimizing

$$\min_{\substack{\mathbf{D}_l \in \mathcal{C}_l \\ \mathbf{D}_h \in \mathbb{R}^{m_h \times p}}} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} \left\| \mathbf{x}_i^h - \mathbf{D}_h \boldsymbol{\alpha}^\star(\mathbf{x}_i^l, \mathbf{D}_l) \right\|_2^2. \tag{3}$$

Pros and Cons

- ☺ testing and training is still consistent;
- ☺ $\mathbf{D}_h$ and $\mathbf{D}_l$ are learned jointly;
- ☹ optimization looks horribly difficult.

Image up-scaling

Scheme with regression

Pipeline:

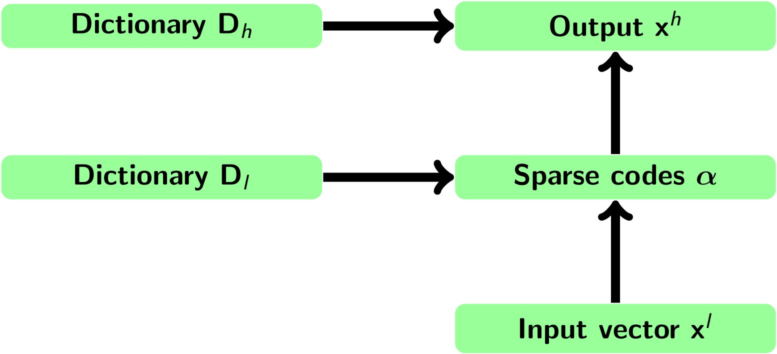


Image up-scaling

Scheme with regression

First step: dictionary learning

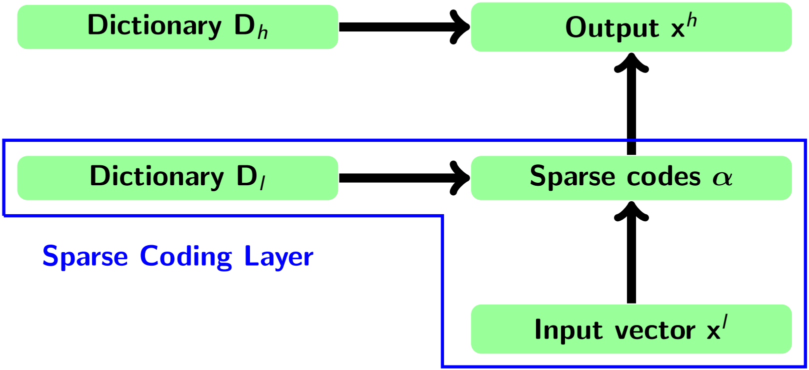


Image up-scaling

Scheme with regression

Second step: regression

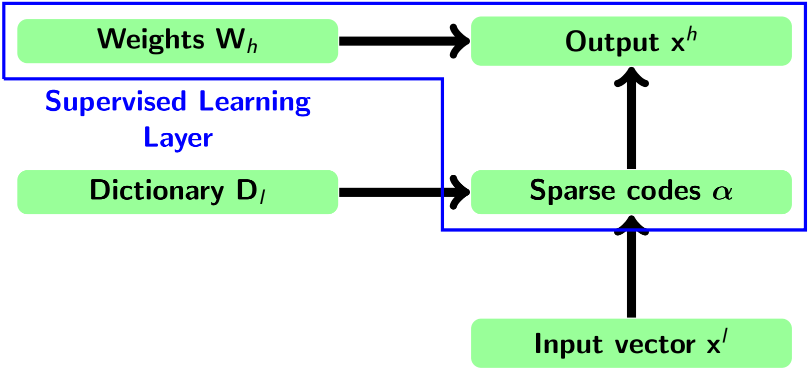
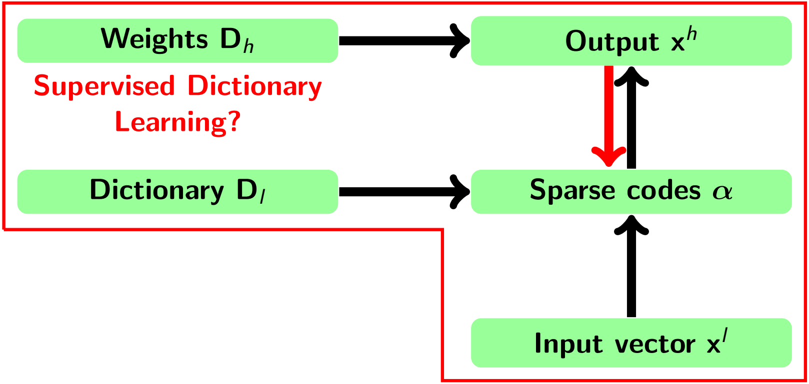


Image up-scaling

Scheme with task-driven dictionary learning

A single step: supervised (task-driven) dictionary learning



In the neural network language, we need **back-propagation** [LeCun et al., 1998].

Image up-scaling

Scheme with task-driven dictionary learning [Couzinie-Devy et al., 2011]

Proposition

In the asymptotic regime, the cost function is differentiable and its gradient admits a simple form [Mairal et al., 2012].

Main recipe of the optimization

- initialize with the regression variant;
- use stochastic gradient descent.
- use classical heuristics from the neural network literature [LeCun et al., 1998].

Image up-scaling

Image from Couzinie-Devy et al. [2011]

Figure: Original

Image up-scaling

Image from Couzinie-Devy et al. [2011]

Figure: Bicubic interpolation

Image up-scaling

Image from Couzinie-Devy et al. [2011]

Figure: from Couzinie-Devy et al. [2011].

Image up-scaling

Image from Couzinie-Devy et al. [2011]

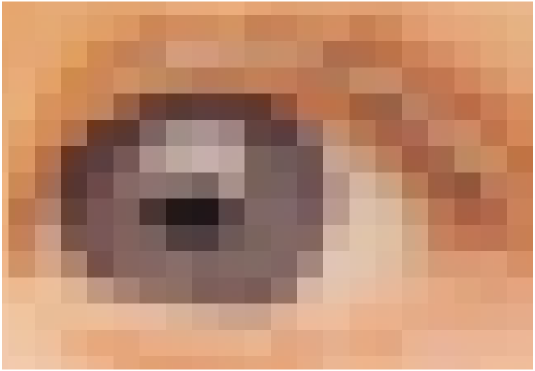


Figure: Original

Image up-scaling

Image from Couzinie-Devy et al. [2011]

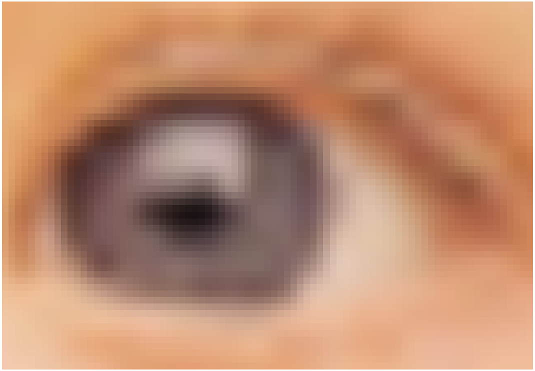


Figure: Bicubic interpolation

Image up-scaling

Image from Couzinie-Devy et al. [2011]

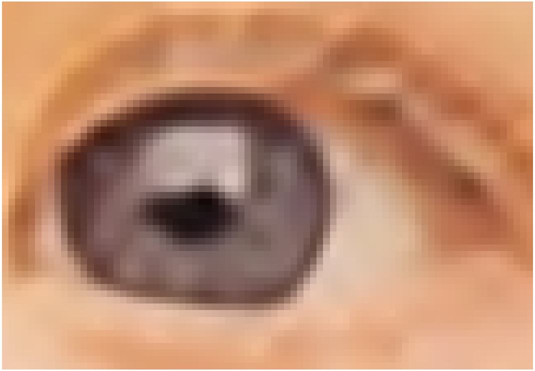


Figure: from Couzinie-Devy et al. [2011].

Image up-scaling

Image from Couzinie-Devy et al. [2011]

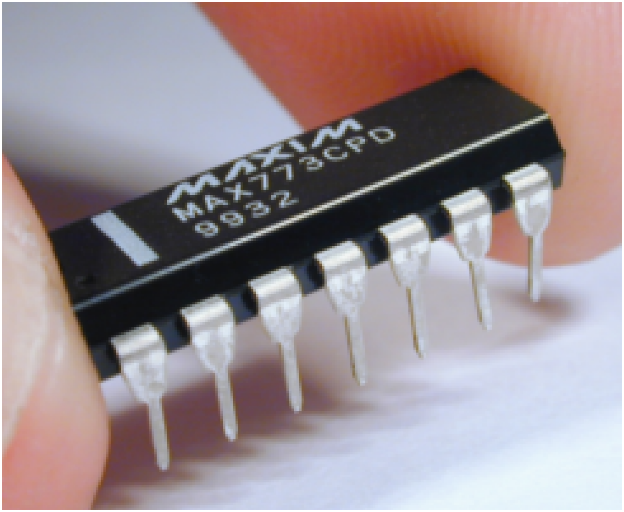


Figure: Original

Image up-scaling

Image from Couzinie-Devy et al. [2011]

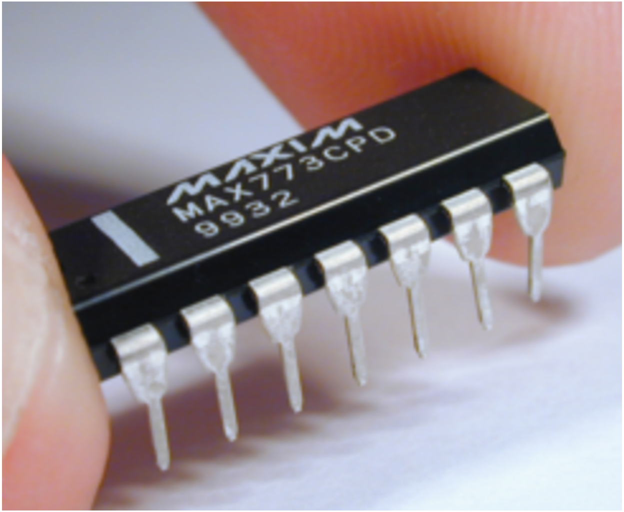


Figure: Bicubic interpolation

Image up-scaling

Image from Couzinie-Devy et al. [2011]

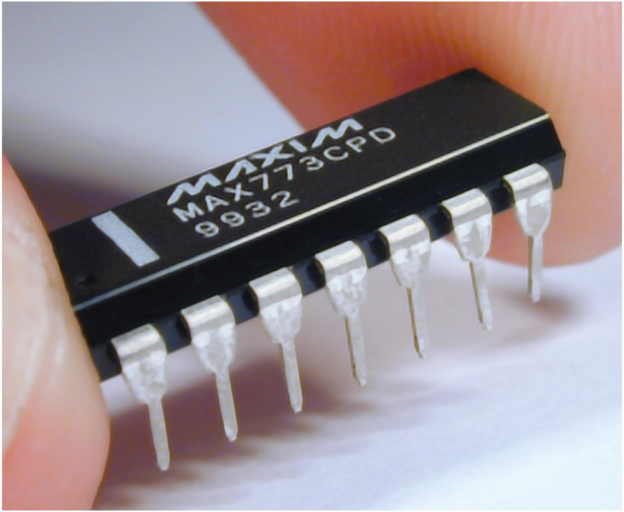


Figure: from Couzinie-Devy et al. [2011].

Image up-scaling

Image from Couzinie-Devy et al. [2011]

Figure: Original

Image up-scaling

Image from Couzinie-Devy et al. [2011]

Figure: Bicubic interpolation

Image up-scaling

Image from Couzinie-Devy et al. [2011]

Figure: from Couzinie-Devy et al. [2011].

Inverting nonlinear local transformations

Remark

The previous up-scaling approaches are generic and can work for other types of local image transformations.

Example: inverse half-toning consists of reconstructing grayscale images from (probably old) binary ones, see, *e.g.*, [Dabov et al., 2006]. A classical algorithm for producing binary images is the one of Floyd and Steinberg [1976].

Inverting nonlinear local transformations

Remark

The previous up-scaling approaches are generic and can work for other types of local image transformations.

Example: inverse half-toning consists of reconstructing grayscale images from (probably old) binary ones, see, *e.g.*, [Dabov et al., 2006]. A classical algorithm for producing binary images is the one of Floyd and Steinberg [1976].

Warning

Inverse half-toning is probably not a hot topic in image processing nowadays.

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Figure: Original

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

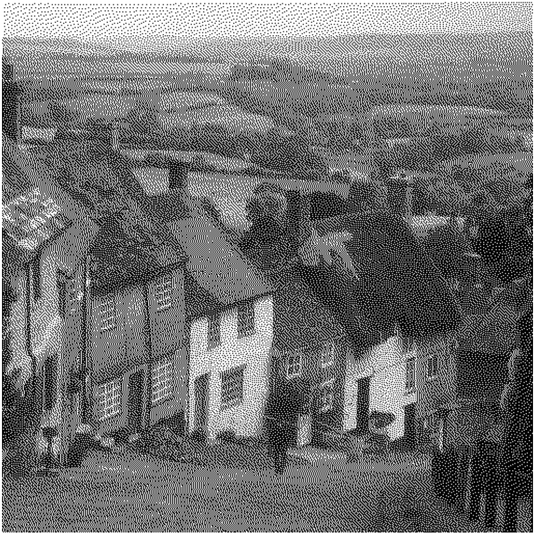


Figure: Binary image

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Figure: Reconstructed.

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Figure: Original

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

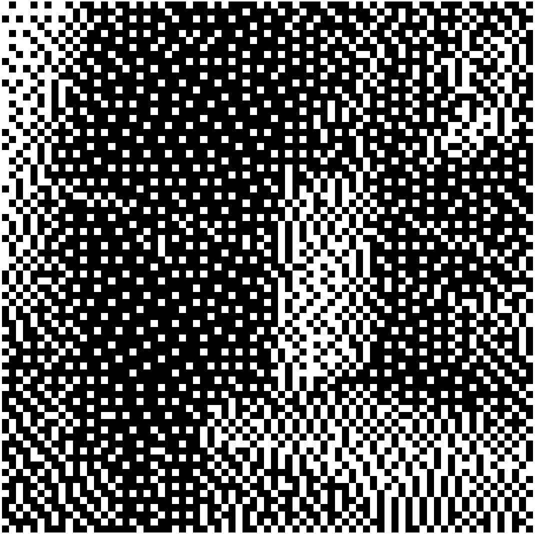


Figure: Binary image

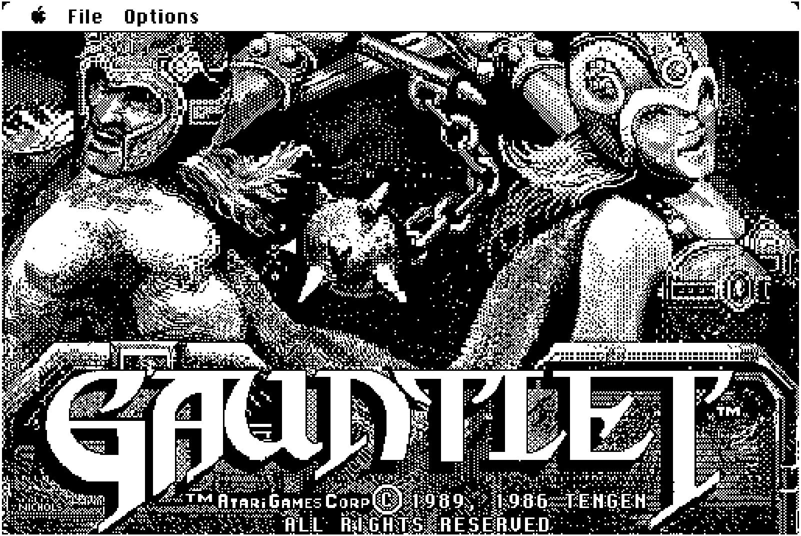
Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Figure: Reconstructed.

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]



Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Inverting nonlinear local transformations

Inverse half-toning [Mairal et al., 2012]

Other patch modeling approaches

Non-local means and non-parametric approaches

Image pixels are well explained by a Nadaraya-Watson estimator:

$$\hat{\mathbf{x}}[i] = \sum_{j=1}^{n} \frac{K_h(\mathbf{y}_i - \mathbf{y}_j)}{\sum_{l=1}^{n} K_h(\mathbf{y}_i - \mathbf{y}_l)} \mathbf{y}[j], \tag{4}$$

with successful application to

- texture synthesis: [Efros and Leung, 1999]
- image denoising **(Non-local means)**: [Buades et al., 2005]
- image demosaicking: [Buades et al., 2009].

Other patch modeling approaches

BM3D

state-of-the-art image denoising approach [Dabov et al., 2007]:

- **block matching**: for each patch, find similar ones in the image;
- **3D wavelet filtering**: denoise blocks of patches with 3D-DCT;
- **patch averaging**: average estimates of overlapping patches;
- **second step with Wiener filtering**: use the first estimate to perform again and improve the previous steps.

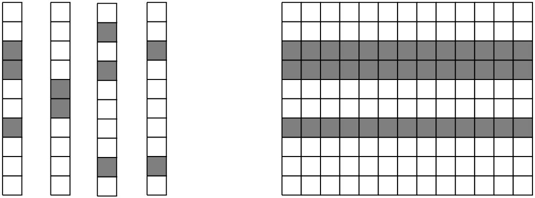
Further refined by Dabov et al. [2009] with shape-adaptive patches and PCA filtering.

Other patch modeling approaches

Non-local sparse models [Mairal et al., 2009b]

Exploit some ideas of BM3D to combine the non-local means principle with dictionary learning.

The main idea is that **similar patches should admit similar decompositions** by using group sparsity:



The approach uses a block matching/clustering step, followed by group sparse coding and patch averaging.

Other patch modeling approaches

Non-local sparse image models

Other patch modeling approaches

Non-local sparse image models

Conclusions from the third part

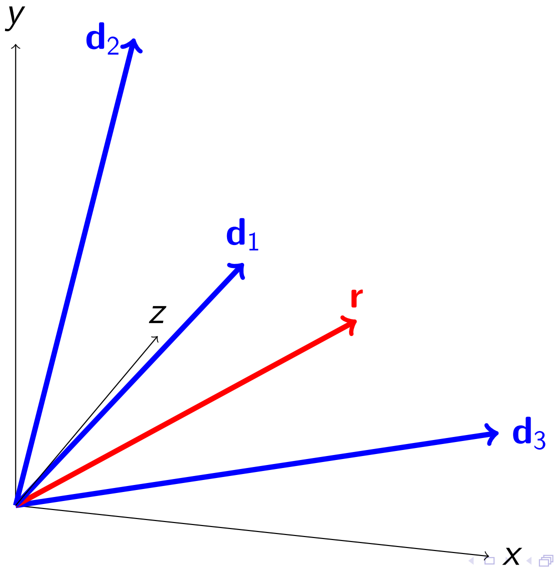
- many inverse problems in image processing can be tackled by modeling natural image patches;
- dictionary learning is one effective way to do it, among others.

Part IV: Optimization for sparse estimation

Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

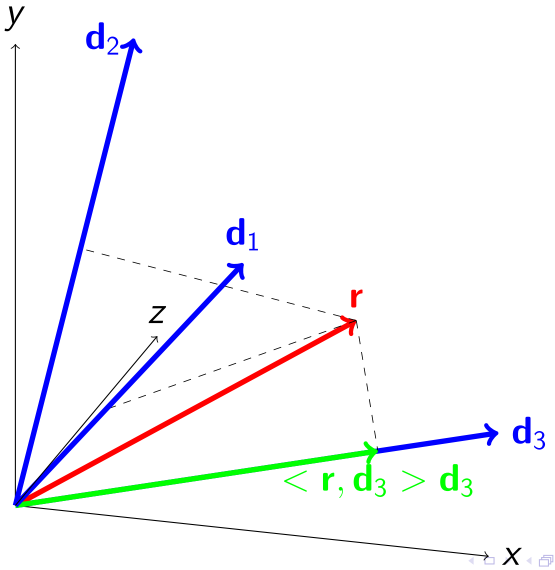
$$\boldsymbol{\alpha} = (0, 0, 0)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

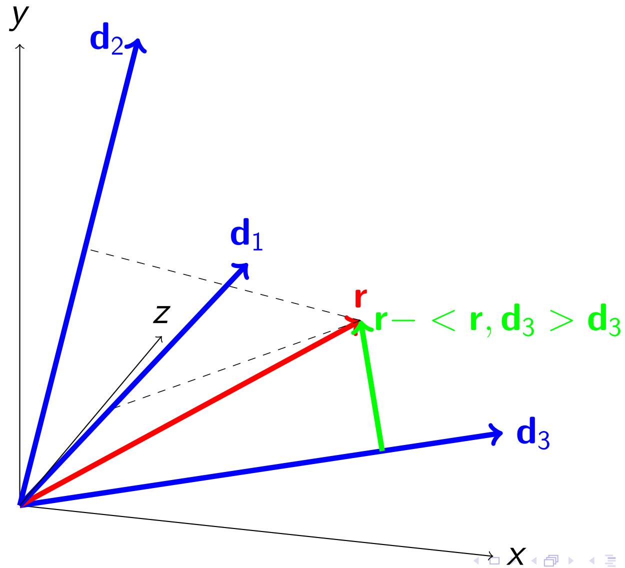
$$\boldsymbol{\alpha} = (0, 0, 0)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

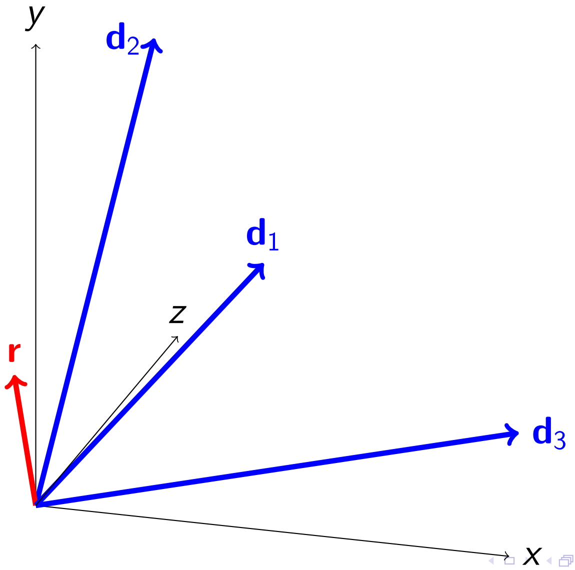
$$\boldsymbol{\alpha} = (0, 0, 0)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

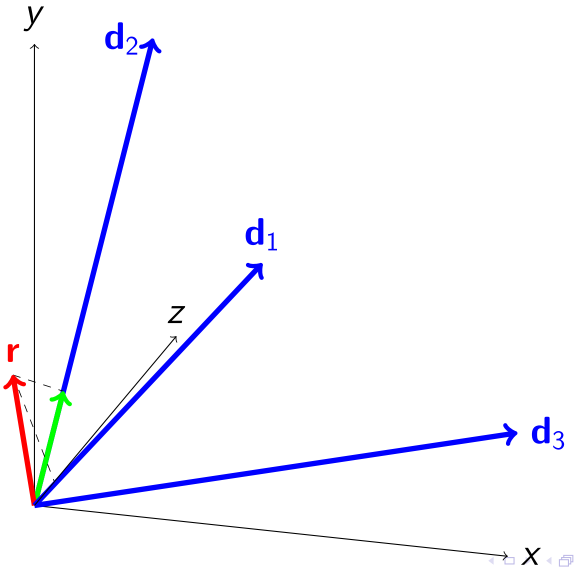
$$\boldsymbol{\alpha} = (0, 0, 0.75)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

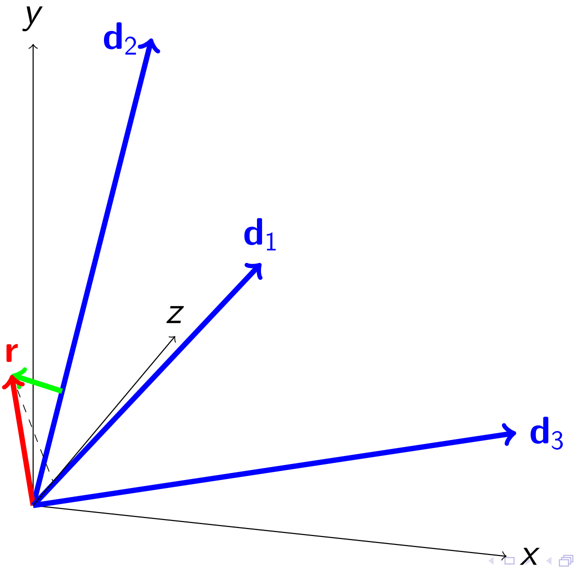
$$\boldsymbol{\alpha} = (0, 0, 0.75)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

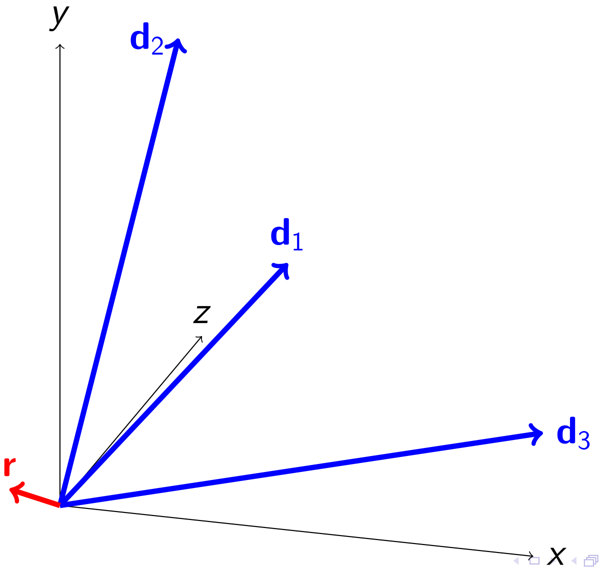
$$\boldsymbol{\alpha} = (0, 0, 0.75)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

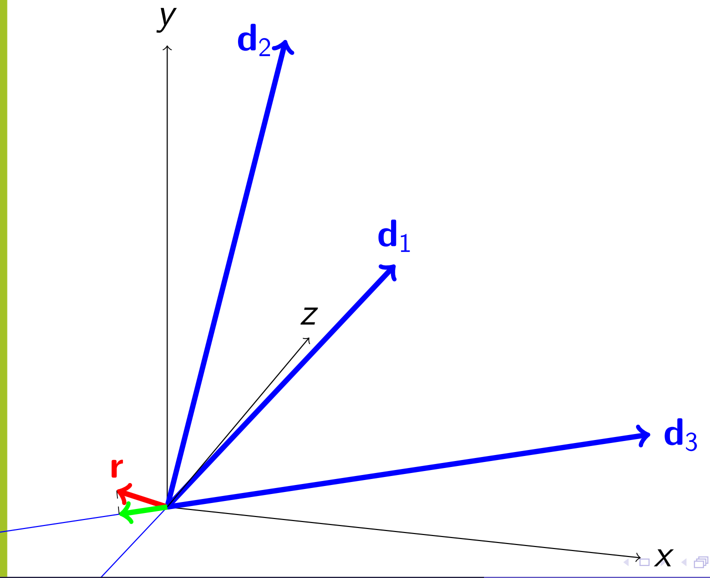
$$\boldsymbol{\alpha} = (0, 0.24, 0.75)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

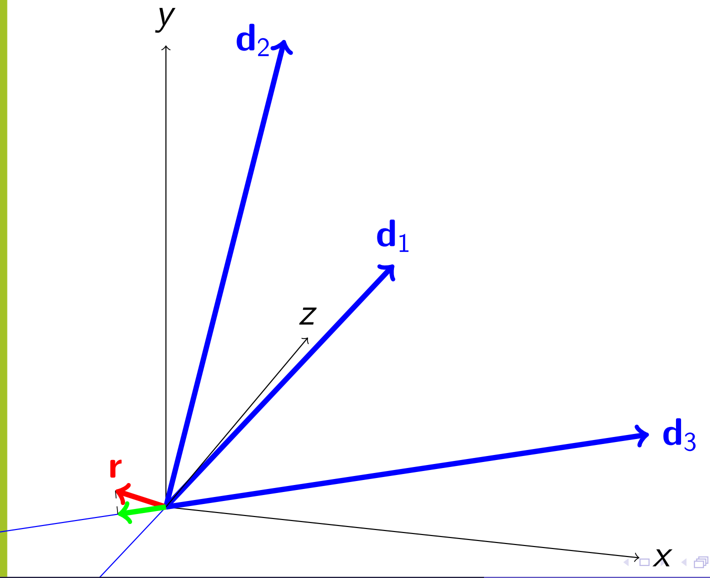
$$\boldsymbol{\alpha} = (0, 0.24, 0.75)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

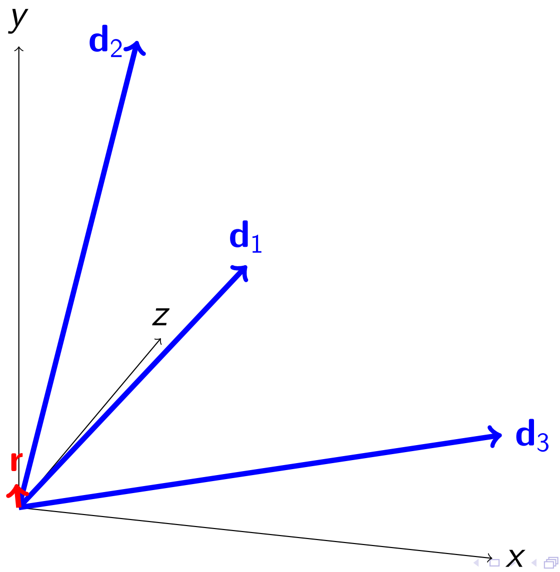
$$\boldsymbol{\alpha} = (0, 0.24, 0.75)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

$$\boldsymbol{\alpha} = (0, 0.24, 0.65)$$



Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \|\underbrace{\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}}_{\mathbf{r}}\|_2^2 \;\text{ s.t. }\; \|\boldsymbol{\alpha}\|_0 \leq k.$$

$\boldsymbol{\alpha} \leftarrow 0$

$\mathbf{r} \leftarrow \mathbf{x}$ (residual).

while $\|\boldsymbol{\alpha}\|_0 < k$ **do**

Select the predictor with maximum inner-product with the residual

$$\hat{\jmath} \leftarrow \arg\max_{j=1,\ldots,p} |\mathbf{d}_j^\top \mathbf{r}|$$

Update the residual and the coefficients

$$\begin{aligned}\boldsymbol{\alpha}[\hat{\jmath}] &\leftarrow \boldsymbol{\alpha}[\hat{\jmath}] + \mathbf{d}_{\hat{\jmath}}^\top \mathbf{r} \\ \mathbf{r} &\leftarrow \mathbf{r} - (\mathbf{d}_{\hat{\jmath}}^\top \mathbf{r})\mathbf{d}_{\hat{\jmath}}\end{aligned}$$

end while

Sparse reconstruction with the ℓ_0-penalty

Matching pursuit [Mallat and Zhang, 1993]

Remarks

- Matching pursuit is a **coordinate descent** algorithm. It greedily selects one coordinate at a time and optimizes the cost function with respect to that coordinate.

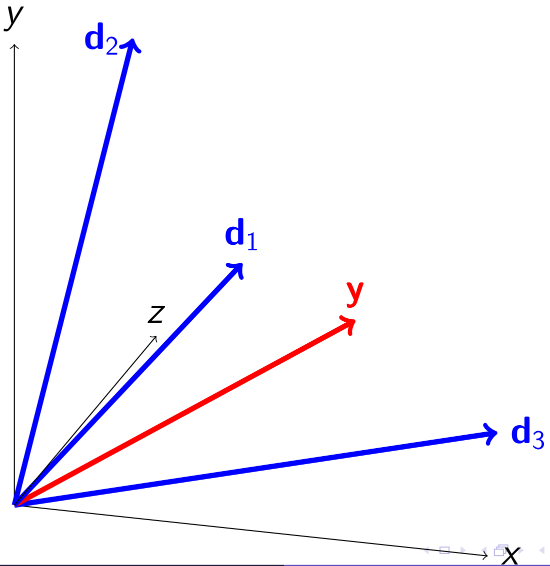
$$\boldsymbol{\alpha}[\hat{\jmath}] \leftarrow \operatorname*{arg\,min}_{\alpha \in \mathbb{R}} \left\| \mathbf{x} - \sum_{l \neq \hat{\jmath}} \boldsymbol{\alpha}[l]\mathbf{d}_l - \alpha \mathbf{d}_{\hat{\jmath}} \right\|_2^2 .$$

- Each coordinate can be selected several times during the process.
- The roots of this algorithm can be found in the statistics literature [Efroymson, 1960].

Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

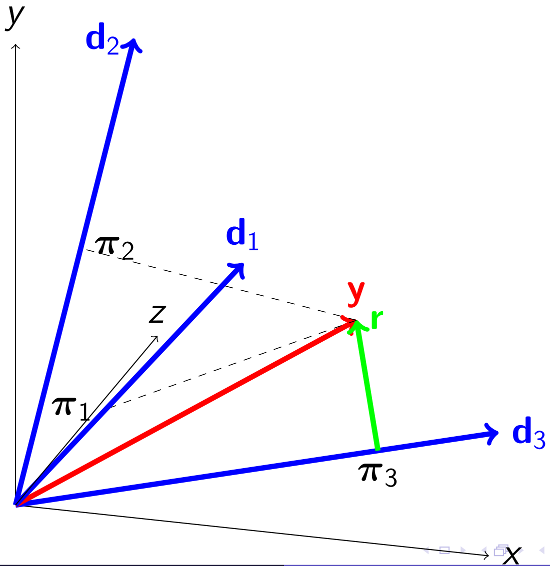
$$\boldsymbol{\alpha} = (0, 0, 0)$$
$$\Gamma = \emptyset$$



Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

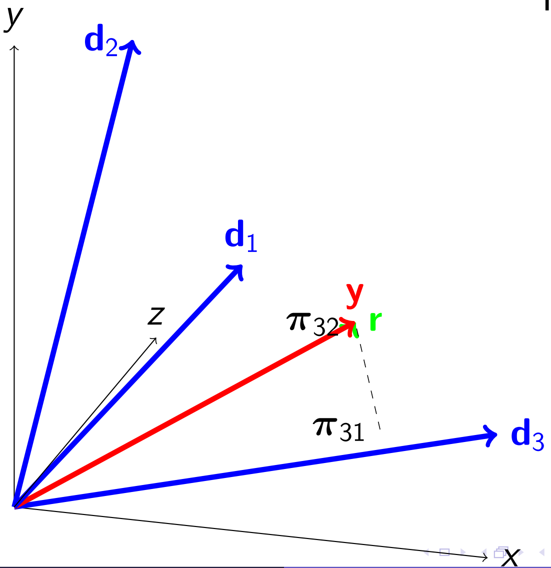
$$\boldsymbol{\alpha} = (0, 0, 0.75)$$
$$\Gamma = \{3\}$$



Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

$$\boldsymbol{\alpha} = (0, 0.29, 0.63)$$
$$\Gamma = \{3, 2\}$$



Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\|_0 \leq k$$

1: $\Gamma = \emptyset$.
2: **for** *iter* $= 1, \ldots, k$ **do**
3: Select the variable that most reduces the objective

$$(\hat{\jmath}, \hat{\boldsymbol{\beta}}) \leftarrow \operatorname*{arg\,min}_{j \in \Gamma^{\complement}, \boldsymbol{\beta}} \|\mathbf{x} - \mathbf{D}_{\Gamma \cup \{j\}} \boldsymbol{\beta}\|_2^2.$$

4: Update the active set: $\Gamma \leftarrow \Gamma \cup \{\hat{\jmath}\}$.
5: Update the coefficients:

$$\boldsymbol{\alpha}[\Gamma] \leftarrow \boldsymbol{\beta} \quad \text{and} \quad \boldsymbol{\alpha}[\Gamma^{\complement}] \leftarrow 0.$$

6: **end for**

Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

Remarks

- this is an **active-set** algorithm.
- when a new variable is selected, the coefficients for the full set Γ are re-optimized:

$$\boldsymbol{\alpha}[\Gamma] = (\mathbf{D}_\Gamma^\top \mathbf{D}_\Gamma)^{-1} \mathbf{D}_\Gamma^\top \mathbf{x},$$

  and the residual is always orthogonal to the matrix $\mathbf{D}_\Gamma$ of previously selected dictionary elements:

$$\mathbf{D}_\Gamma^\top (\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}) = \mathbf{D}_\Gamma^\top (\mathbf{x} - \mathbf{D}_\Gamma \boldsymbol{\alpha}[\Gamma]) = 0.$$

- several variants of OMP exist regarding the selection rule of $\hat{\jmath}$. The one we use appears in Cotter et al. [1999].

Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

Keys for a fast implementation

- If available, use the Gram matrix $\mathbf{G} = \mathbf{D}^\top \mathbf{D}$;
- Maintain the computation of $\mathbf{D}^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\alpha})$,
- Update the Cholesky decomposition of $(\mathbf{D}_\Gamma^\top \mathbf{D}_\Gamma)^{-1}$.

The total complexity for decomposing n k-sparse signals of size m with a dictionary of size p is

$$\underbrace{O(p^2 m)}_{\text{Gram matrix}} + \underbrace{O(nk^3)}_{\text{Cholesky}} + \underbrace{O(n(pm + pk^2))}_{\mathbf{D}^\top(\mathbf{x}-\mathbf{D}\boldsymbol{\alpha})} = O(np(m + k^2))$$

It is also possible to use the matrix inversion lemma instead of a Cholesky decomposition.

Sparse reconstruction with the ℓ_0-penalty

Orthogonal matching pursuit [Pati et al., 1993]

Example with the software SPAMS

Software available at `http://spams-devel.gforge.inria.fr/`.

```
>> I=double(imread('data/lena.eps'))/255;
>> %extract all patches of I
>> X=im2col(I,[8 8],'sliding');
>> %load a dictionary of size 64 x 256
>> D=load('dict.mat');
>>
>> %set the sparsity parameter L to 10
>> param.L=10;
>> alpha=mexOMP(X,D,param);
```

On this dual-core laptop: **110000 signals processed per second!**

Sparse reconstruction with the ℓ_0-penalty

Iterative hard-thresholding [Herrity et al., 2006, Blumensath and Davies, 2009]

Require: Signal $\mathbf{x}$ in $\mathbb{R}^m$, dictionary $\mathbf{D}$ in $\mathbb{R}^{m \times p}$, target sparsity k, gradient descent step size η, number of iterations T.

Initialize $\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha}_0$;

for $t = 1, \ldots, T$ **do**

perform one step of gradient descent:

$$\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} + \eta \mathbf{D}^\top (\mathbf{x} - \mathbf{D}\boldsymbol{\alpha});$$

choose τ to be the k-th largest entry of $\{|\boldsymbol{\alpha}[1]|, \ldots, |\boldsymbol{\alpha}[p]|\}$;

for $j = 1, \ldots, p$ **do**

hard-thresholding:

$$\boldsymbol{\alpha}[j] \leftarrow \begin{cases} \boldsymbol{\alpha}[j] & \text{if } |\boldsymbol{\alpha}[j]| \geq \tau \\ 0 & \text{otherwise.} \end{cases}$$

end for

end for

return the sparse decomposition $\boldsymbol{\alpha}$ in $\mathbb{R}^p$.

Sparse reconstruction with the ℓ_0-penalty

Iterative hard-thresholding [Herrity et al., 2006, Blumensath and Davies, 2009]

Remarks

- This is a **projected gradient algorithm**;

$$\boldsymbol{\alpha} \leftarrow \Pi_{\|.\|_0 \leq k} \left[\boldsymbol{\alpha} - \eta \nabla f(\boldsymbol{\alpha}) \right].$$

  It performs one gradient descent step, followed by a Euclidean projection onto the non-convex set of k-sparse vectors.

- it can be easily extended to the (approximate) minimization of

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_0.$$

  In that case, it is as a **proximal gradient algorithm**.

- it can be seen to iteratively decreases the value of the objective function from the **majorization-minimization** point of view.

Sparse reconstruction with the ℓ_0-penalty

Majorization-minimization principle [Lange et al., 2000]

The principle for (approximately) minimizing a general cost function f:

$$\min_{\boldsymbol{\alpha} \in \mathcal{A}} f(\boldsymbol{\alpha}).$$

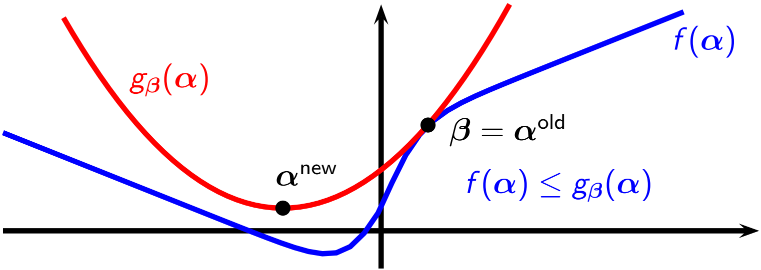


Figure: At each step, we update $\boldsymbol{\alpha} \in \arg\min_{\boldsymbol{\alpha} \in \mathcal{A}} g_{\boldsymbol{\beta}}(\boldsymbol{\alpha})$

Sparse reconstruction with the ℓ_0-penalty

Majorization-minimization principle [Lange et al., 2000]

The principle for (approximately) minimizing a general cost function f:

$$\min_{\boldsymbol{\alpha} \in \mathcal{A}} f(\boldsymbol{\alpha}).$$

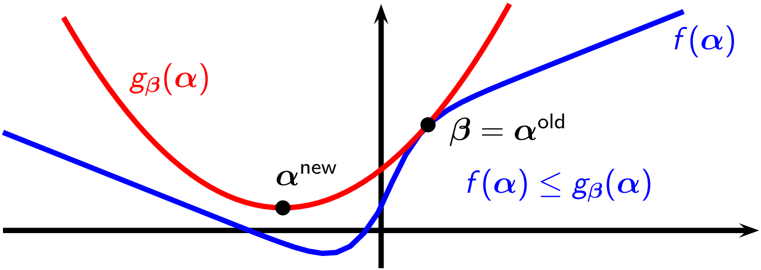


Figure: At each step, we update $\boldsymbol{\alpha} \in \arg\min_{\boldsymbol{\alpha} \in \mathcal{A}} g_{\boldsymbol{\beta}}(\boldsymbol{\alpha})$

What is the surrogate for the iterative hard-thresholding algorithm?

Sparse reconstruction with the ℓ_0-penalty

Majorization-minimization principle [Lange et al., 2000]

The principle for (approximately) minimizing a general cost function f:

$$\min_{\boldsymbol{\alpha} \in \mathcal{A}} f(\boldsymbol{\alpha}).$$

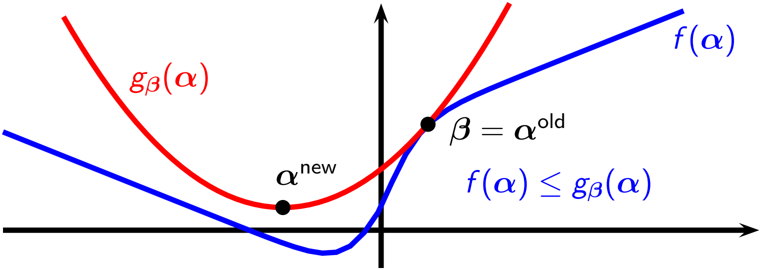
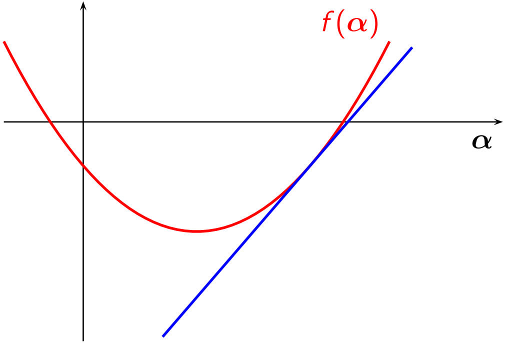


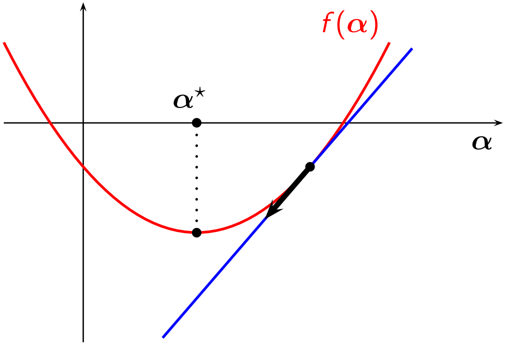
Figure: At each step, we update $\boldsymbol{\alpha} \in \arg\min_{\boldsymbol{\alpha} \in \mathcal{A}} g_{\boldsymbol{\beta}}(\boldsymbol{\alpha})$

We need to introduce a few principles first...

Why do we care about convexity?



Local observations give information about the global optimum

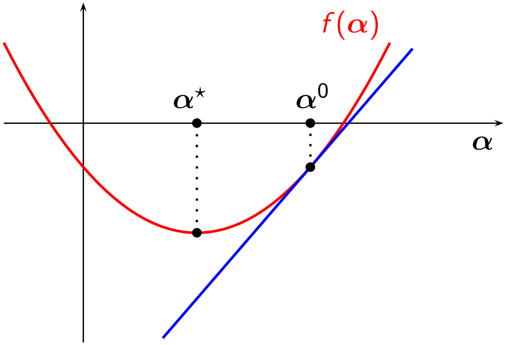


- $\nabla f(\alpha) = 0$ is a necessary and sufficient optimality condition for differentiable convex functions;
- it is often easy to upper-bound $f(\alpha) - f^\star$.

Introduction of a few optimization principles

An important inequality for smooth convex functions

If f is convex

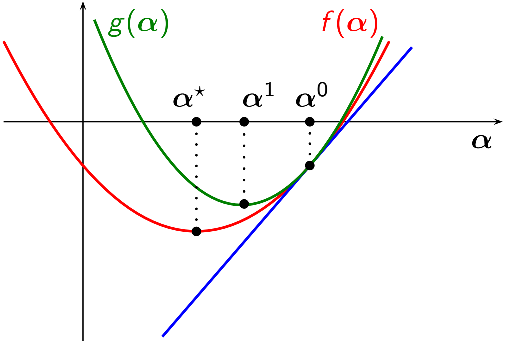


- $f(\boldsymbol{\alpha}) \geq \underbrace{f(\boldsymbol{\alpha}^0) + \nabla f(\boldsymbol{\alpha}^0)^\top(\boldsymbol{\alpha} - \boldsymbol{\alpha}^0)}_{\text{linear approximation}}$;
- this is an equivalent definition of convexity for smooth functions.

Introduction of a few optimization principles

An important inequality for smooth functions

If ∇f is L-Lipschitz continuous (f does not need to be convex)

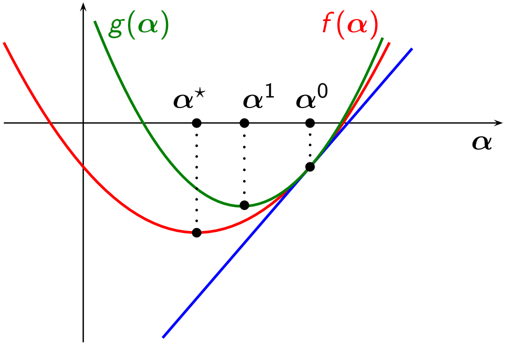


- $f(\boldsymbol{\alpha}) \leq g(\boldsymbol{\alpha}) = \underbrace{f(\boldsymbol{\alpha}^0) + \nabla f(\boldsymbol{\alpha}^0)^\top(\boldsymbol{\alpha} - \boldsymbol{\alpha}^0)}_{\text{linear approximation}} + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|_2^2;$

Introduction of a few optimization principles

An important inequality for smooth functions

If ∇f is L-Lipschitz continuous (f does not need to be convex)

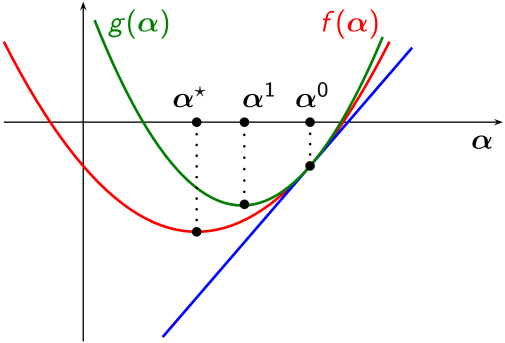


- $f(\boldsymbol{\alpha}) \leq g(\boldsymbol{\alpha}) = \underbrace{f(\boldsymbol{\alpha}^0) + \nabla f(\boldsymbol{\alpha}^0)^\top(\boldsymbol{\alpha} - \boldsymbol{\alpha}^0)}_{\text{linear approximation}} + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|_2^2$;
- $g(\boldsymbol{\alpha}) = C_{\boldsymbol{\alpha}^0} + \frac{L}{2}\|\boldsymbol{\alpha}^0 - (1/L)\nabla f(\boldsymbol{\alpha}^0) - \boldsymbol{\alpha}\|_2^2$.

Introduction of a few optimization principles

An important inequality for smooth functions

If ∇f is L-Lipschitz continuous (f does not need to be convex)



- $f(\boldsymbol{\alpha}) \leq g(\boldsymbol{\alpha}) = \underbrace{f(\boldsymbol{\alpha}^0) + \nabla f(\boldsymbol{\alpha}^0)^\top(\boldsymbol{\alpha} - \boldsymbol{\alpha}^0)}_{\text{linear approximation}} + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|_2^2$;
- $\boldsymbol{\alpha}^1 = \boldsymbol{\alpha}^0 - \frac{1}{L}\nabla f(\boldsymbol{\alpha}^0)$. (gradient descent step).

Introduction of a few optimization principles

Gradient Descent Algorithm

Assume that f is convex and differentiable, and that ∇f is L-Lipschitz.

Theorem

Consider the algorithm

$$\boxed{\boldsymbol{\alpha}^t \leftarrow \boldsymbol{\alpha}^{t-1} - \frac{1}{L}\nabla f(\boldsymbol{\alpha}^{t-1}).}$$

Then,

$$f(\boldsymbol{\alpha}^t) - f^\star \leq \frac{L\|\boldsymbol{\alpha}^0 - \boldsymbol{\alpha}^\star\|_2^2}{2t}.$$

Remarks

- the convergence rate improves under additional assumptions on f (strong convexity);
- some variants have a $O(1/t^2)$ convergence rate [Nesterov, 2004].

Proof (1/2)

Proof of the main inequality for smooth functions

We want to show that for all $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$,

$$f(\boldsymbol{\alpha}) \leq f(\boldsymbol{\beta}) + \nabla f(\boldsymbol{\beta})^\top(\boldsymbol{\alpha} - \boldsymbol{\beta}) + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2^2.$$

By using Taylor's theorem with integral form,

$$f(\boldsymbol{\alpha}) - f(\boldsymbol{\beta}) = \int_0^1 \nabla f(t\boldsymbol{\alpha} + (1-t)\boldsymbol{\beta})^\top(\boldsymbol{\alpha} - \boldsymbol{\beta})dt.$$

Then,

$$\begin{aligned}
f(\boldsymbol{\alpha}) - f(\boldsymbol{\beta}) - \nabla f(\boldsymbol{\beta})^\top(\boldsymbol{\alpha} - \boldsymbol{\beta}) &\leq \int_0^1 (\nabla f(t\boldsymbol{\alpha} + (1-t)\boldsymbol{\beta}) - \nabla f(\boldsymbol{\beta}))^\top(\boldsymbol{\alpha} - \boldsymbol{\beta})dt \\
&\leq \int_0^1 |(\nabla f(t\boldsymbol{\alpha} + (1-t)\boldsymbol{\beta}) - \nabla f(\boldsymbol{\beta}))^\top(\boldsymbol{\alpha} - \boldsymbol{\beta})|dt \\
&\leq \int_0^1 \|\nabla f(t\boldsymbol{\alpha} + (1-t)\boldsymbol{\beta}) - \nabla f(\boldsymbol{\beta})\|_2\|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2 dt \quad \text{(C.-S.)} \\
&\leq \int_0^1 Lt\|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2^2 dt = \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2^2.
\end{aligned}$$

Proof (2/2)

Proof of the theorem

We have shown that for all $\boldsymbol{\alpha}$,

$$f(\boldsymbol{\alpha}) \leq g_t(\boldsymbol{\alpha}) = f(\boldsymbol{\alpha}^{t-1}) + \nabla f(\boldsymbol{\alpha}^{t-1})^\top(\boldsymbol{\alpha} - \boldsymbol{\alpha}^{t-1}) + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^{t-1}\|_2^2.$$

g_t is minimized by $\boldsymbol{\alpha}^t$; it can be rewritten $g_t(\boldsymbol{\alpha}) = g_t(\boldsymbol{\alpha}^t) + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^t\|_2^2$. Then,

$$\begin{aligned}
f(\boldsymbol{\alpha}^t) \leq g_t(\boldsymbol{\alpha}^t) &= g_t(\boldsymbol{\alpha}^\star) - \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^t\|_2^2 \\
&= f(\boldsymbol{\alpha}^{t-1}) + \nabla f(\boldsymbol{\alpha}^{t-1})^\top(\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^{t-1}) + \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^{t-1}\|_2^2 - \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^t\|_2^2 \\
&\leq f^\star + \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^{t-1}\|_2^2 - \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^t\|_2^2.
\end{aligned}$$

By summing from $t = 1$ to T, we have a telescopic sum

$$T(f(\boldsymbol{\alpha}^T) - f^\star) \leq \sum_{t=1}^{T} f(\boldsymbol{\alpha}^t) - f^\star \leq \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^0\|_2^2 - \frac{L}{2}\|\boldsymbol{\alpha}^\star - \boldsymbol{\alpha}^T\|_2^2.$$

Sparse reconstruction with the ℓ_0-penalty

iterative hard-thresholding [Herrity et al., 2006, Blumensath and Davies, 2009]

What is the surrogate $g_\beta(\alpha)$?

Sparse reconstruction with the ℓ_0-penalty

iterative hard-thresholding [Herrity et al., 2006, Blumensath and Davies, 2009]

Simply the same as for the gradient descent algorithm:

$$g_{\boldsymbol{\beta}}(\boldsymbol{\alpha}) \triangleq f(\boldsymbol{\alpha}) + \nabla f(\boldsymbol{\beta})^\top(\boldsymbol{\alpha} - \boldsymbol{\beta}) + \frac{L}{2}\|\boldsymbol{\beta} - \boldsymbol{\alpha}\|_2^2,$$

with $\boldsymbol{\beta} = \boldsymbol{\alpha}^{\text{old}}$, $L = (1/\eta)$ and $f(\boldsymbol{\alpha}) = (1/2)\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2$. Indeed,

$$g_{\boldsymbol{\beta}}(\boldsymbol{\alpha}) = C_{\boldsymbol{\beta}} + \frac{L}{2}\|\boldsymbol{\beta} + \eta\mathbf{D}^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\beta}) - \boldsymbol{\alpha}\|_2^2.$$

and the update can be rewritten

$$\begin{aligned}\boldsymbol{\alpha} \leftarrow & \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in\mathbb{R}^p : \|\boldsymbol{\alpha}\|_0 \leq k} g_{\boldsymbol{\beta}}(\boldsymbol{\alpha}) \\ &= \Pi_{\|.\|_0 \leq k}\left[\boldsymbol{\beta} + \eta\mathbf{D}^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\beta})\right].\end{aligned}$$

Sparse reconstruction with the ℓ_1-norm

For the ℓ_0-penalty, we have seen

1. a **coordinate descent** algorithm (matching pursuit);
2. a **gradient descent** algorithm (iterative hard-thresholding);
3. an **active-set** algorithm (orthogonal matching pursuit);

For ℓ_1, the same three classes of methods play an important role.

Sparse reconstruction with the ℓ_1-norm

Projected gradient descent

Suppose we want to solve

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \;\; \text{s.t.} \;\; \|\boldsymbol{\alpha}_1\|_1 \leq \mu.$$

Sparse reconstruction with the ℓ_1-norm

Projected gradient descent

Suppose we want to solve

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}_1\|_1 \leq \mu.$$

The following update with η small enough converges to a solution

$$\boldsymbol{\alpha} \leftarrow \Pi_{\|.\|_1 \leq \mu} \left[\boldsymbol{\alpha} + \eta \mathbf{D}^\top (\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}) \right].$$

Sparse reconstruction with the ℓ_1-norm

Projected gradient descent

Suppose we want to solve

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{\alpha}_1\|_1 \leq \mu.$$

The following update with η small enough converges to a solution

$$\boldsymbol{\alpha} \leftarrow \Pi_{\|.\|_1 \leq \mu}\left[\boldsymbol{\alpha} + \eta \mathbf{D}^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\alpha})\right].$$

Remarks

- the convergence rate is the same as the gradient descent method for smooth convex functions;
- when L is unknown, efficient line-search scheme can be used.
- the principle is the same as for the iterative hard-thresholding algorithm.

see [Nesterov, 2004, Bertsekas, 1999, Boyd and Vandenberghe, 2004].

Sparse reconstruction with the ℓ_1-norm

The proximal gradient method

We consider a smooth convex function f and a non-smooth regularizer ψ.

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} f(\boldsymbol{\alpha}) + \psi(\boldsymbol{\alpha})$$

For example,

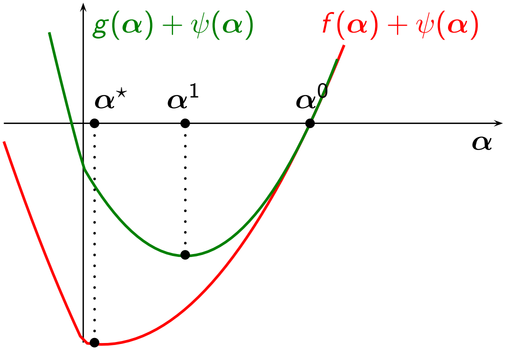
$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1.$$

- the objective function is not differentiable.
- an extension of gradient descent for such a problem is called **"proximal gradient descent"** [Beck and Teboulle, 2009, Nesterov, 2013].

Sparse reconstruction with the ℓ_1-norm

An important inequality for composite functions

If ∇f is L-Lipschitz continuous



- $f(\alpha) + \psi(\alpha) \leq$
  $f(\alpha^0) + \nabla f(\alpha^0)^\top(\alpha - \alpha^0) + \frac{L}{2}\|\alpha - \alpha^0\|_2^2 + \psi(\alpha)$;
- α^1 minimizes $g + \psi$.

Sparse reconstruction with the ℓ_1-norm

The proximal gradient method

Gradient descent for minimizing f consists of

$$\boldsymbol{\alpha}^t \leftarrow \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in\mathbb{R}^p} g_t(\boldsymbol{\alpha}) \quad \Longleftrightarrow \quad \boldsymbol{\alpha}^t \leftarrow \boldsymbol{\alpha}^{t-1} - \frac{1}{L}\nabla f(\boldsymbol{\alpha}^{t-1}).$$

The proximal gradient method for minimizing $f + \psi$ consists of

$$\boldsymbol{\alpha}^t \leftarrow \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in\mathbb{R}^p} g_t(\boldsymbol{\alpha}) + \psi(\boldsymbol{\alpha}),$$

which is equivalent to

$$\boxed{\boldsymbol{\alpha}^t \leftarrow \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in\mathbb{R}^p} \frac{1}{2}\left\|\boldsymbol{\alpha}^{t-1} - \frac{1}{L}\nabla f(\boldsymbol{\alpha}^{t-1}) - \boldsymbol{\alpha}\right\|_2^2 + \frac{1}{L}\psi(\boldsymbol{\alpha}).}$$

It requires computing efficiently the **proximal operator** of ψ.

$$\boldsymbol{\alpha} \mapsto \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in\mathbb{R}^p} \ \frac{1}{2}\|\boldsymbol{\beta} - \boldsymbol{\alpha}\|_2^2 + \psi(\boldsymbol{\alpha}).$$

Sparse reconstruction with the ℓ_1-norm

The proximal gradient method

Remarks

- also known as forward-backward algorithm;
- has similar convergence rates as the gradient descent method.
- there exists line search schemes to automatically tune L;
- there exists accelerated schemes [Beck and Teboulle, 2009, Nesterov, 2013].

The case of ℓ_1

The proximal operator of $\lambda\|.\|_1$ is the soft-thresholding operator

$$\boldsymbol{\alpha}[j] = \text{sign}(\boldsymbol{\beta}[j])(|\boldsymbol{\beta}[j]| - \lambda)^+.$$

The resulting algorithm is called **iterative soft-thresholding** [Nowak and Figueiredo, 2001, Figueiredo and Nowak, 2003, Starck et al., 2003, Daubechies et al., 2004].

The proximal operator for the group Lasso penalty

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\boldsymbol{\beta} - \boldsymbol{\alpha}\|_2^2 + \lambda \sum_{g \in \mathcal{G}} \|\boldsymbol{\alpha}[g]\|_q.$$

For $q = 2$,

$$\boldsymbol{\alpha}[g] = \frac{\boldsymbol{\beta}[g]}{\|\boldsymbol{\beta}[g]\|_2}(\|\boldsymbol{\beta}[g]\|_2 - \lambda)^+, \quad \forall g \in \mathcal{G}.$$

For $q = \infty$,

$$\boldsymbol{\alpha}[g] = \boldsymbol{\beta}[g] - \Pi_{\|.\|_1 \leq \lambda}[\boldsymbol{\beta}[g]], \quad \forall g \in \mathcal{G}.$$

These formula generalize soft-thresholding to groups of variables.

Sparse reconstruction with the ℓ_1-norm

The proximal gradient method

A few proximal operators:

- ℓ_0-penalty: hard-thresholding;
- ℓ_1-norm: soft-thresholding;
- group-Lasso: group soft-thresholding;
- fused-lasso (1D total variation): [Hoefling, 2010];
- **hierarchical norms**: [Jenatton et al., 2011b], $O(p)$ complexity;
- **overlapping group Lasso with ℓ_∞-norm**: [Mairal et al., 2010b], (link with network flow optimization);

Sparse reconstruction with the ℓ_1-norm

Coordinate descent for the Lasso [Fu, 1998]

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1.$$

The coordinate descent method consists of iteratively fixing all variables and optimizing with respect to one:

$$\boldsymbol{\alpha}[j] \leftarrow \arg\min_{\alpha \in \mathbb{R}} \frac{1}{2}\| \underbrace{\mathbf{x} - \sum_{l \neq j} \boldsymbol{\alpha}[l]\mathbf{d}_l}_{\mathbf{r}} - \alpha\mathbf{d}_j\|_2^2 + \lambda|\alpha|.$$

Assume the columns of $\mathbf{D}$ to have unit ℓ_2-norm,

$$\boldsymbol{\alpha}_j \leftarrow \text{sign}(\mathbf{d}_j^\top \mathbf{r})(|\mathbf{d}_j^\top \mathbf{r}| - \lambda)^+$$

This involves again the **soft-thresholding** operator.

Sparse reconstruction with the ℓ_1-norm

Coordinate descent for the Lasso [Fu, 1998]

Remarks

- no parameter to tune!
- several strategies are possible for selecting the variable to update.
- impressive performance with five lines of code.
- coordinate descent + nonsmooth objective is **not convergent in general**. Here, the problem is equivalent to a convex smooth optimization problem with separable constraints

$$\min_{\boldsymbol{\alpha}_+, \boldsymbol{\alpha}_-} \frac{1}{2}\|\mathbf{x} - \mathbf{D}_+\boldsymbol{\alpha}_+ + \mathbf{D}_-\boldsymbol{\alpha}_-\|_2^2 + \lambda \boldsymbol{\alpha}_+^T \mathbf{1} + \lambda \boldsymbol{\alpha}_-^T \mathbf{1} \ \text{ s.t. } \ \boldsymbol{\alpha}_-, \boldsymbol{\alpha}_+ \geq 0.$$

  For this specific problem, the algorithm is **convergent**.
- can be extended to group-Lasso, or other loss functions.
- j can be picked up at random, or by cycling (harder to analyze).

Sparse reconstruction with the ℓ_1-norm

Smoothing techniques: reweighted ℓ_2 [Daubechies et al., 2010, Bach et al., 2012]

Let us start from something simple

$$a^2 - 2ab + b^2 \geq 0.$$

Sparse reconstruction with the ℓ_1-norm

Smoothing techniques: reweighted ℓ_2 [Daubechies et al., 2010, Bach et al., 2012]

Let us start from something simple

$$a^2 - 2ab + b^2 \geq 0.$$

Then

$$a \leq \frac{1}{2}\left(\frac{a^2}{b} + b\right) \text{ with equality iff } a = b$$

and

$$\|\boldsymbol{\alpha}\|_1 = \min_{\eta_j \geq 0} \frac{1}{2} \sum_{j=1}^{p} \frac{\boldsymbol{\alpha}[j]^2}{\eta_j} + \eta_j.$$

The formulation becomes

$$\min_{\boldsymbol{\alpha}, \eta_j \geq \varepsilon} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \frac{\lambda}{2} \sum_{j=1}^{p} \frac{\boldsymbol{\alpha}[j]^2}{\eta_j} + \eta_j.$$

Sparse reconstruction with the ℓ_1-norm

Homotopy [Osborne et al., 2000b, Efron et al., 2004, Ritter, 1962]

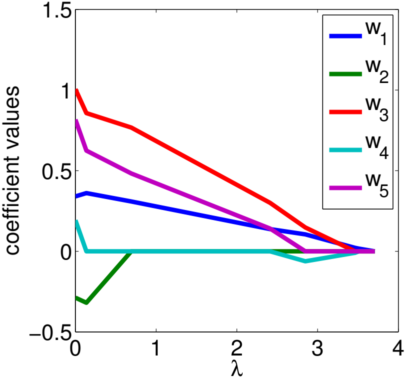


Figure: The regularization path of the Lasso is piecewise linear.

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1.$$

property discoved by Markowitz [1952].

Sparse reconstruction with the ℓ_1-norm

Homotopy [Osborne et al., 2000b, Efron et al., 2004, Ritter, 1962]

Theorem

$\boldsymbol{\alpha}$ is a solution of the Lasso if and only if

$$\begin{cases} |\mathbf{d}_j^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\alpha})| & \leq & \lambda \ \text{ if } \boldsymbol{\alpha}[j] = 0 \\ \mathbf{d}_j^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}) & = & \lambda \operatorname{sign}(\boldsymbol{\alpha}[j]) \ \text{ otherwise.} \end{cases}$$

Consequence

$$\boldsymbol{\alpha}^\star[\Gamma] = (\mathbf{D}_\Gamma^T \mathbf{D}_\Gamma)^{-1}(\mathbf{D}_\Gamma^T \mathbf{x} - \lambda \operatorname{sign}(\boldsymbol{\alpha}^\star[\Gamma])) = \mathbf{A} + \lambda \mathbf{B},$$

where $\Gamma = \{j \ \text{ s.t. } \ \boldsymbol{\alpha}[j] \neq 0\}$. If we know Γ and the signs of $\boldsymbol{\alpha}^\star$ in advance, we have a closed form solution.

Following the piecewise linear regularization path is called the **homotopy** method [Osborne et al., 2000a, Efron et al., 2004].

Sparse reconstruction with the ℓ_1-norm

Homotopy [Osborne et al., 2000b, Efron et al., 2004, Ritter, 1962]

The regularization path $(\lambda, \boldsymbol{\alpha}^\star(\lambda))$ is piecewise linear.

1. Start from the trivial solution $(\lambda = \|\mathbf{D}^T\mathbf{x}\|_\infty, \boldsymbol{\alpha}^\star(\lambda) = 0)$.
2. Define $\Gamma = \{j \text{ s.t. } |\mathbf{d}_j^\top \mathbf{x}| = \lambda\}$,
3. Follow the regularization path: $\boldsymbol{\alpha}^\star_\Gamma(\lambda) = \mathbf{A} + \lambda\mathbf{B}$, keeping $\boldsymbol{\alpha}^\star_{\Gamma^c} = 0$, decreasing the value of λ, until one of the following event occurs:
   - $\exists j \notin \Gamma$ such that $|\mathbf{d}_j^\top(\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}^\star(\lambda))| = \lambda$, then $\Gamma \leftarrow \Gamma \cup \{j\}$.
   - $\exists j \in \Gamma$ such that $\boldsymbol{\alpha}^\star(\lambda) = 0$, then $\Gamma \leftarrow \Gamma \setminus \{j\}$.
4. Update the direction of the path and go back to 3.

Hidden assumptions

- the regularization path is unique.
- variables enter the path one at a time.

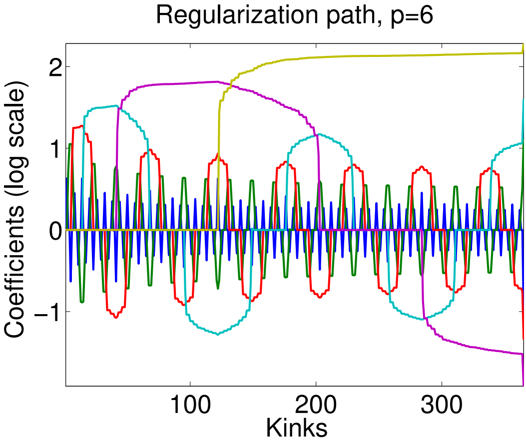
Extremely efficient for **small/medium scale** problems ($p \leq 10\,000$) and/or **very sparse** problems (when implemented correctly). **Robust to correlated features**. Can solve the elastic-net.

Sparse reconstruction with the ℓ_1-norm

Homotopy [Osborne et al., 2000b, Efron et al., 2004, Ritter, 1962]

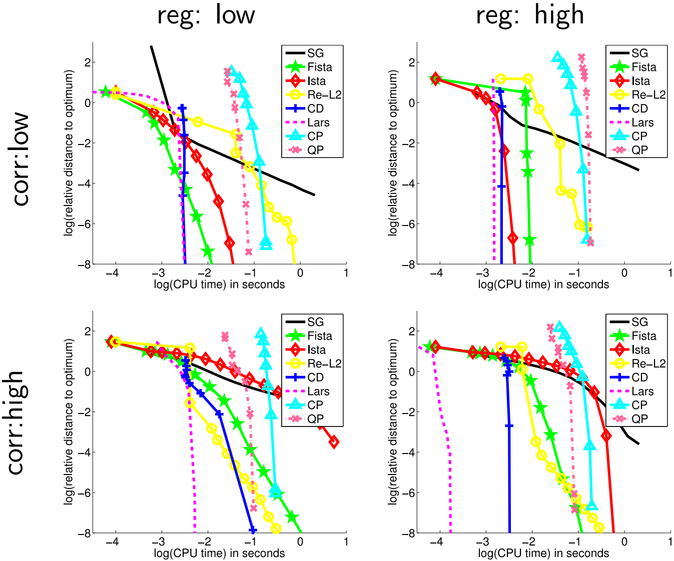
Theorem - worst case analysis [Mairal and Yu, 2012]

In the worst-case, the regularization path of the Lasso has exactly $(3^p + 1)/2$ *linear segments.*



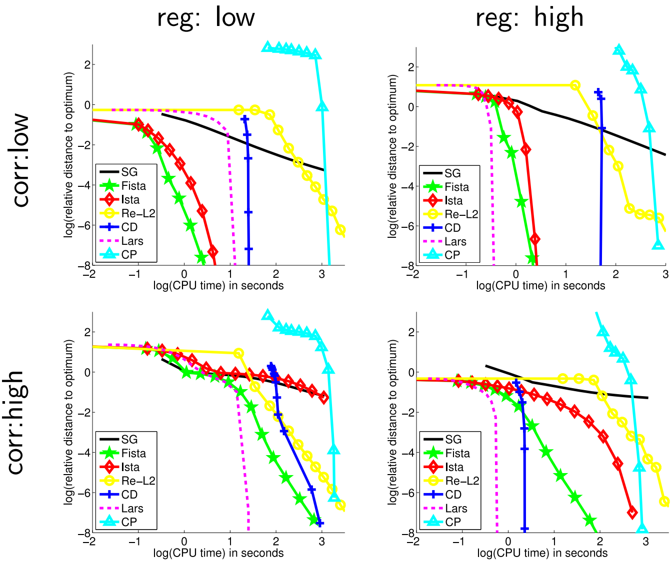
Sparse reconstruction with the ℓ_1-norm

Lasso empirical comparison: Lasso, small scale ($n = 200, p = 200$)



Sparse reconstruction with the ℓ_1-norm

Empirical comparison: Lasso, medium scale ($n = 2000, p = 10000$)



Sparse reconstruction with the ℓ_1-norm

Empirical comparison: conclusions

Lasso

- Generic methods (subgradient descent, QP/CP solvers) are slow;
- homotopy fastest in **low dimension** and/or for **high correlation**
- Proximal methods are competitive
  - esp. larger setting and/or weak corr. and/or weak reg. and/or low precision
- Coordinate descent
  - usually dominated by LARS;
  - but much simpler to implement!

Smooth Losses and other regularization

- LARS not available $\rightarrow$ (block) coordinate descent, proximal gradient methods are good candidates.

Iterative reweighted ℓ_1-algorithms

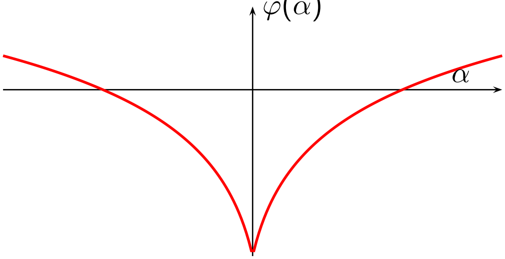
DC (difference of convex) - Programming

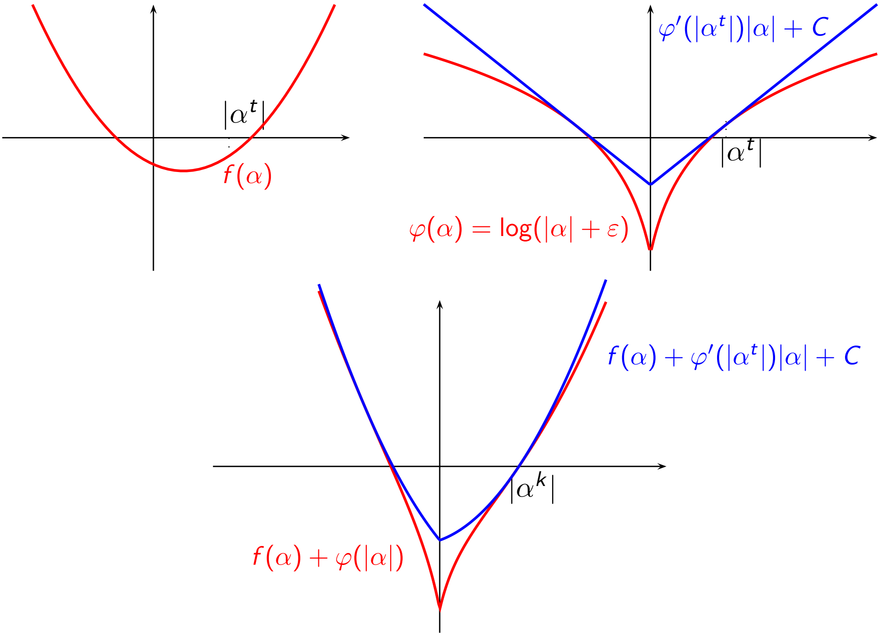
Remember? Concave functions with a kink at zero

$\psi(\boldsymbol{\alpha}) = \sum_{j=1}^{p} \varphi(|\boldsymbol{\alpha}[j]|)$.

- ℓ_q-"pseudo-norm", with $0 < q < 1$: $\psi(\mathbf{w}) \triangleq \sum_{j=1}^{p} (|\boldsymbol{\alpha}[j]| + \varepsilon)^q$,
- log penalty, $\psi(\mathbf{w}) \triangleq \sum_{j=1}^{p} \log(|\boldsymbol{\alpha}[j]| + \varepsilon)$,

φ is any function that looks like this:





$|\alpha^t|$
$f(\alpha)$
$\varphi'(|\alpha^t|)|\alpha| + C$
$|\alpha^t|$
$\varphi(\alpha) = \log(|\alpha| + \varepsilon)$
$f(\alpha) + \varphi'(|\alpha^t|)|\alpha| + C$
$|\alpha^k|$
$f(\alpha) + \varphi(|\alpha|)$

DC (difference of convex) - Programming

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} f(\boldsymbol{\alpha}) + \lambda \sum_{j=1}^{p} \varphi(|\boldsymbol{\alpha}[j]|).$$

This problem is non-convex. f is convex, and φ is concave on $\mathbb{R}^+$.
if $\boldsymbol{\alpha}^k$ is the current estimate at iteration t, the algorithm solves

$$\boldsymbol{\alpha}^{t+1} \leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \left[f(\boldsymbol{\alpha}) + \lambda \sum_{j=1}^{p} \varphi'(|\boldsymbol{\alpha}^t[j]|)|\boldsymbol{\alpha}[j]| \right],$$

which is a **reweighted**-ℓ_1 problem [Figueiredo and Nowak, 2005, Figueiredo et al., 2007, Candès et al., 2008].

Warning: It does not solve the non-convex problem, only provides a stationary point.

In practice, each iteration sets to zero small coefficients. After $2-3$ iterations, the result does not change much.

Optimization for Dictionary Learning

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^{p \times n} \\ \mathbf{D} \in \mathcal{C}}} \sum_{i=1}^{n} \frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \psi(\boldsymbol{\alpha}_i)$$

$$\mathcal{C} \triangleq \{\mathbf{D} \in \mathbb{R}^{m \times p} \ \text{ s.t. } \ \forall j = 1, \ldots, p, \quad \|\mathbf{d}_j\|_2 \leq 1\}.$$

Classical approach

- Alternate minimization between $\mathbf{D}$ and $\boldsymbol{\alpha}$ (MOD with $\psi = \ell_0$ [Engan et al., 1999], K-SVD with $\psi = \ell_0$ [Aharon et al., 2006], [Lee et al., 2007] with $\psi = \ell_1$);
- good results, reliable, but can be slow when n is large!

Optimization for Dictionary Learning

Empirical risk minimization point of view

$$\min_{\mathbf{D}\in\mathcal{C}} f_n(\mathbf{D}) = \min_{\mathbf{D}\in\mathcal{C}} \frac{1}{n}\sum_{i=1}^{n} L(\mathbf{x}_i, \mathbf{D}),$$

where

$$L(\mathbf{x}, \mathbf{D}) \triangleq \min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\psi(\boldsymbol{\alpha}).$$

Which formulation are we interested in?

$$\min_{\mathbf{D}\in\mathcal{C}} \left\{ f(\mathbf{D}) = \mathbb{E}_{\mathbf{x}}[L(\mathbf{x}, \mathbf{D})] \approx \lim_{n\to+\infty} \frac{1}{n}\sum_{i=1}^{n} L(\mathbf{x}_i, \mathbf{D}) \right\}$$

[Bottou and Bousquet, 2008]: Online learning can

- handle potentially infinite or dynamic datasets,
- be dramatically faster than batch algorithms.

Optimization for Dictionary Learning

Stochastic gradient descent

Recipe

- draw a single point $\mathbf{x}_t$ (or a mini-batch) at each iteration;
- update

$$\mathbf{D} \leftarrow \Pi_{\mathcal{C}}[\mathbf{D} - \eta_t \nabla_{\mathbf{D}} L(\mathbf{x}_t, \mathbf{D})],$$

which is equivalent (up to some assumptions) to

$$\boldsymbol{\alpha}_t \leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1,$$

$$\mathbf{D} \leftarrow \Pi_{\mathcal{C}}[\mathbf{D} + \eta_t(\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)\boldsymbol{\alpha}_t^\top].$$

Optimization for Dictionary Learning

Stochastic gradient descent

Recipe

- draw a single point $\mathbf{x}_t$ (or a mini-batch) at each iteration;
- update

$$\mathbf{D} \leftarrow \Pi_{\mathcal{C}}[\mathbf{D} - \eta_t \nabla_{\mathbf{D}} L(\mathbf{x}_t, \mathbf{D})],$$

  which is equivalent (up to some assumptions) to

$$\boldsymbol{\alpha}_t \leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1,$$

$$\mathbf{D} \leftarrow \Pi_{\mathcal{C}}[\mathbf{D} + \eta_t(\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)\boldsymbol{\alpha}_t^\top].$$

Remark

- historically, this is very close to the original algorithm of Olshausen and Field [1996].

Optimization for Dictionary Learning

Stochastic gradient descent

Recipe

- draw a single point $\mathbf{x}_t$ (or a mini-batch) at each iteration;
- update

$$\mathbf{D} \leftarrow \Pi_{\mathcal{C}}[\mathbf{D} - \eta_t \nabla_{\mathbf{D}} L(\mathbf{x}_t, \mathbf{D})],$$

which is equivalent (up to some assumptions) to

$$\boldsymbol{\alpha}_t \leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1,$$

$$\mathbf{D} \leftarrow \Pi_{\mathcal{C}}[\mathbf{D} + \eta_t(\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)\boldsymbol{\alpha}_t^\top].$$

Pros and cons

☺ can be effective in practice;

☹ difficult to tune.

Optimization for Dictionary Learning

Online dictionary learning [Mairal et al., 2010a]

Recipe

- stochastic majorization-minimization algorithm;
- relies on a fast dictionary update;
- easier to tune (the implementation of SPAMS has been successfully used by others in plenty of "exotic" unexpected scenarios.

Optimization for Dictionary Learning

Online dictionary learning [Mairal et al., 2010a]

Require: $\mathbf{D}_0 \in \mathbb{R}^{m \times p}$ (initial dictionary); $\lambda \in \mathbb{R}$

$\mathbf{C}_0 = 0$, $\mathbf{B}_0 = 0$.

for t=1,...,T **do**

  Draw $\mathbf{x}_t$

  Sparse Coding: $\boldsymbol{\alpha}_t \leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}_{t-1}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1,$

  Aggregate sufficient statistics

  $\mathbf{C}_t \leftarrow \mathbf{C}_{t-1} + \boldsymbol{\alpha}_t\boldsymbol{\alpha}_t^T$, $\mathbf{B}_t \leftarrow \mathbf{B}_{t-1} + \mathbf{x}_t\boldsymbol{\alpha}_t^T$

  Dictionary update

$$\begin{aligned}\mathbf{D}_t &\leftarrow \arg\min_{\mathbf{D} \in \mathcal{C}} \frac{1}{t}\sum_{i=1}^{t}\Big(\frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\Big). \\ &= \arg\min_{\mathbf{D} \in \mathcal{C}} \frac{1}{t}\Big(\frac{1}{2}\operatorname{Tr}(\mathbf{D}^T\mathbf{D}\mathbf{C}_t) - \operatorname{Tr}(\mathbf{D}^T\mathbf{B}_t)\Big).\end{aligned}$$

end for

Optimization for Dictionary Learning

Fast dictionary udpate [Mairal et al., 2010a]

Require: $\mathbf{D}_0 \in \mathcal{C}$ (input dictionary); $\mathbf{X} \in \mathbb{R}^{m \times n}$ (dataset); $\mathbf{A} \in \mathbb{R}^{p \times n}$ (sparse codes);

Initialization: $\mathbf{D} \leftarrow \mathbf{D}_0$; $\mathbf{B} \leftarrow \mathbf{X}\mathbf{A}^\top$; $\mathbf{C} \leftarrow \mathbf{A}\mathbf{A}^\top$;

repeat

  for $j = 1, \ldots, p$ **do**

    update the j-th column:

$$\mathbf{d}_j \leftarrow \frac{1}{\mathbf{C}[j,j]}(\mathbf{b}_j - \mathbf{D}\mathbf{c}_j) + \mathbf{d}_j,$$

$$\mathbf{d}_j \leftarrow \frac{1}{\max(\|\mathbf{d}_j\|_2, 1)}\mathbf{d}_j.$$

  end for

until convergence;

return $\mathbf{D}$ (updated dictionary).

Optimization for Dictionary Learning

Fast dictionary udpate [Mairal et al., 2010a]

Minimizing with respect to one column $\mathbf{d}_j$ when keeping the other columns fixed can be formulated as

$$\mathbf{d}_j \leftarrow \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2\leq 1}{\arg\min} \left[\sum_{i=1}^{n} \frac{1}{2} \left\| \mathbf{x}_i - \sum_{l\neq j} \boldsymbol{\alpha}_i[l]\mathbf{d}_l - \boldsymbol{\alpha}_i[j]\mathbf{d} \right\|_2^2 \right].$$

Optimization for Dictionary Learning

Fast dictionary udpate [Mairal et al., 2010a]

Minimizing with respect to one column $\mathbf{d}_j$ when keeping the other columns fixed can be formulated as

$$\mathbf{d}_j \leftarrow \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2\leq 1}{\arg\min} \left[\sum_{i=1}^{n} \frac{1}{2} \left\| \mathbf{x}_i - \sum_{l\neq j} \boldsymbol{\alpha}_i[l]\mathbf{d}_l - \boldsymbol{\alpha}_i[j]\mathbf{d} \right\|_2^2 \right].$$

Then, in a matrix form

$$\mathbf{d}_j \leftarrow \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2\leq 1}{\arg\min} \left[\frac{1}{2} \left\| \mathbf{X} - \mathbf{DA} + \mathbf{d}_j \boldsymbol{\alpha}^j - \mathbf{d}\boldsymbol{\alpha}^j \right\|_F^2 \right],$$

Optimization for Dictionary Learning

Fast dictionary udpate [Mairal et al., 2010a]

Minimizing with respect to one column $\mathbf{d}_j$ when keeping the other columns fixed can be formulated as

$$\mathbf{d}_j \leftarrow \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2 \leq 1}{\arg\min} \left[\sum_{i=1}^{n} \frac{1}{2} \left\| \mathbf{x}_i - \sum_{l \neq j} \boldsymbol{\alpha}_i[l]\mathbf{d}_l - \boldsymbol{\alpha}_i[j]\mathbf{d} \right\|_2^2 \right].$$

After expanding the Frobenius norm and removing the constant term,

$$\begin{aligned}
\mathbf{d}_j \leftarrow & \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2 \leq 1}{\arg\min} \left[-\mathbf{d}^\top (\mathbf{X} - \mathbf{D}\mathbf{A} + \mathbf{d}_j \boldsymbol{\alpha}^j) \boldsymbol{\alpha}^{j\top} + \frac{1}{2} \left\| \mathbf{d}\boldsymbol{\alpha}^j \right\|_\mathsf{F}^2 \right] \\
= & \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2 \leq 1}{\arg\min} \left[-\mathbf{d}^\top (\mathbf{b}_j - \mathbf{D}\mathbf{c}_j + \mathbf{d}_j \mathbf{C}[j,j]) + \frac{1}{2} \|\mathbf{d}\|_2^2 \, \mathbf{C}[j,j] \right] \\
= & \underset{\mathbf{d}\in\mathbb{R}^m, \|\mathbf{d}\|_2 \leq 1}{\arg\min} \left[\frac{1}{2} \left\| \frac{1}{\mathbf{C}[j,j]} (\mathbf{b}_j - \mathbf{D}\mathbf{c}_j) + \mathbf{d}_j - \mathbf{d} \right\|_2^2 \right],
\end{aligned}$$

Optimization for Dictionary Learning

[Mairal et al., 2010a]

Which guarantees do we have?

Under a few reasonable assumptions,

- we build a surrogate function $\hat{g}_t$ of the expected cost f verifying

$$\lim_{t \to +\infty} \hat{g}_t(\mathbf{D}_t) - f(\mathbf{D}_t) = 0,$$

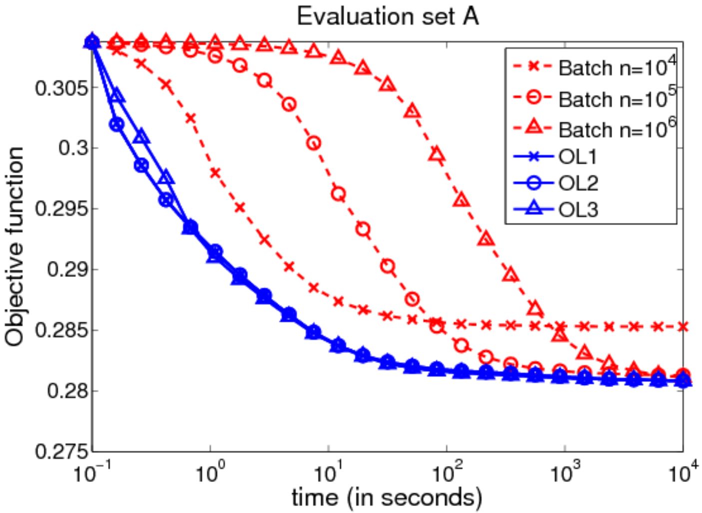
- $\mathbf{D}_t$ is asymptotically close to a stationary point.

Extensions (all implemented in SPAMS)

- non-negative matrix decompositions;
- sparse PCA (sparse dictionaries);
- fused-lasso regularizations (piecewise constant dictionaries);
- non-convex regularization, structured regularization.

Optimization for Dictionary Learning

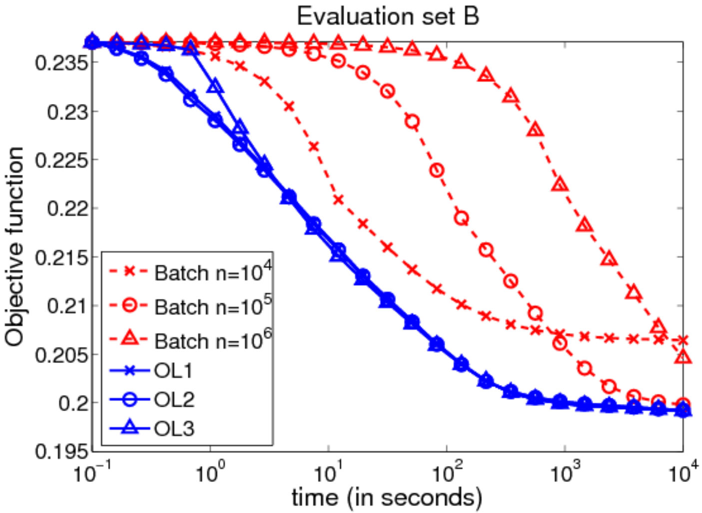
Experimental results, batch vs online



$m = 8 \times 8, \ p = 256$

Optimization for Dictionary Learning

Experimental results, batch vs online



$$m = 12 \times 12 \times 3, \; p = 512$$

Conclusions from the fourth part

- there are a few algorithms for sparse estimation that are efficient and easy to implement;
- there is no algorithm that wins all the time;
- designing an evaluation benchmark that makes sense is **hard**.

Conclusions from the fourth part

- there are a few algorithms for sparse estimation that are efficient and easy to implement;
- there is no algorithm that wins all the time;
- designing an evaluation benchmark that makes sense is **hard**.

What was not covered

- stochastic optimization for sparse estimation;
- proximal splitting algorithms.

Advertisement again

- the SPAMS toolbox already contains lots of code (C++ interfaced with Matlab, Python, R) for learning dictionaries, factorizing matrices (NMF, archetypal analysis), solving sparse estimation problems, including most of the algorithms we have presented. `http://spams-devel.gforge.inria.fr/`.

Part V: Application cases

Application cases

Case 1

- use of dictionary learning for processing electrophysiological data from the visual cortex.

Case 2

- use of structured sparse models for next-generation DNA/RNA sequencing.

References I

M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.

N. Ahmed, T. Natarajan, and K. R. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, 100(1):90–93, 1974.

H. Akaike. Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, volume 1, pages 267–281, 1973.

F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundation and Trends in Machine Learning*, 4:1–106, 2012.

S. Bakin. *Adaptive regression and model selection in data mining problems*. PhD thesis, 1999.

References II

R. G. Baraniuk, V. Cevher, M. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4): 1982–2001, 2010.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

A. J. Bell and T. J. Sejnowski. The "independent components" of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of the ACM SIGGRAPH Conference*, 2000.

D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, 1999. 2nd edition.

References III

T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

J.M. Borwein and A.S. Lewis. *Convex analysis and nonlinear optimization: theory and examples*. Springer, 2006.

T. Bossomaier and A. W. Snyder. Why spatial frequency processing in the visual cortex? *Vision Research*, 26(8):1307–1309, 1986.

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

A. Buades, B. Coll, and J.-M. Morel. A review of image denoising algorithms, with a new one. *SIAM Journal on Multiscale Modeling and Simulation*, 4(2):490–530, 2005.

References IV

A. Buades, B. Coll, J.-M. Morel, and C. Sbert. Self-similarity driven color demosaicking. *IEEE Transactions on Image Processing*, 18(6): 1192–1202, 2009.

P. Bühlmann and S. Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer, 2011.

T. T. Cai. Adaptive wavelet estimation: a block thresholding and oracle inequality approach. *Annals of Statistics*, 27(3):898–924, 1999.

E. J. Candès and D. L. Donoho. Recovering edges in ill-posed inverse problems: Optimality of curvelet frames. *Annals of Statistics*, 30(3): 784–842, 2002.

E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2): 489–509, 2006.

References V

E.J. Candès, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications*, 14(5): 877–905, 2008.

M. Carandini, J. B. Demb, V. Mante, D. J. Tolhurst, Y. Dan, B. A. Olshausen, J. L. Gallant, and N. C. Rust. Do we know what the early visual system does? *The Journal of Neuroscience*, 25(46): 10577–10597, 2005.

J.-F. Cardoso. Dependence, correlation and Gaussianity in independent component analysis. *Journal of Machine Learning Research*, 4: 1177–1203, 2003.

V. Cehver, M. Duarte, C. Hedge, and R.G. Baraniuk. Sparse signal recovery using markov random fields. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

References VI

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1999.

J. F. Claerbout and F. Muir. Robust modeling with erratic data. *Geophysics*, 38(5):826–844, 1973.

S. F. Cotter, J. Adler, B. Rao, and K. Kreutz-Delgado. Forward sequential algorithms for best basis selection. In *IEEE Proceedings of Vision Image and Signal Processing*, pages 235–244, 1999.

F. Couzinie-Devy, J. Mairal, F. Bach, and J. Ponce. Dictionary learning for deblurring and digital zoom. *preprint arXiv:1110.0957*, 2011.

T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley and Sons, 2006. 2nd edition.

A. Cutler and L. Breiman. Archetypal analysis. *Technometrics*, 36(4): 338–347, 1994.

References VII

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Inverse halftoning by pointwise shape-adaptive DCT regularized deconvolution. In *Proceedings of the International TICSP Workshop on Spectral Methods Multirate Signal Processing (SMMSP)*, 2006.

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. BM3D image denoising with shape-adaptive principal component analysis. In *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*, 2009.

I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11): 1413–1457, 2004.

References VIII

I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1):1–38, 2010.

J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A*, 2(7):1160–1169, 1985.

M. Do and M. Vertterli. *Contourlets, Beyond Wavelets*. Academic Press, 2003.

D. L. Donoho and J. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.

References IX

A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 1999.

M. A. Efroymson. Multiple regression analysis. *Mathematical methods for digital computers*, 9(1):191–203, 1960.

M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

K. Engan, S. O. Aase, and J. H. Husoy. Method of optimal directions for frame design. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1999.

M. Fazel, H. Hindi, and S. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *American Control Conference*, volume 6, pages 4734–4739, 2001.

References X

M. A. T. Figueiredo and R. D. Nowak. An EM algorithm for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 12(8):906–916, 2003.

M. A. T. Figueiredo and R. D. Nowak. A bound optimization approach to wavelet-based image deconvolution. In *Proceedings of the International Conference on Image Processing (ICIP)*, 2005.

M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 16(12): 2980–2991, 2007.

R. W. Floyd and L. Steinberg. An adaptive algorithm for spatial grey scale. In *Proceedings of the Society of Information Display*, volume 17, pages 75–77, 1976.

I. E Frank and J. H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.

References XI

W.J. Fu. Penalized regressions: the bridge versus the lasso. *Journal of Computational and Graphical Statistics*, 7(3):397–416, 1998.

G. M. Furnival and R. W. Wilson. Regressions by leaps and bounds. *Technometrics*, 16(4):499–511, 1974.

A. Gersho and R. M. Gray. *Vector quantization and signal compression*. Kluwer Academic Publishers, 1992.

Y. Grandvalet and S. Canu. Outcomes of the equivalence of adaptive ridge with least absolute shrinkage. In *Advances in Neural Information Processing Systems (NIPS)*, 1999.

H. Grassmann. LXXXVII. On the theory of compound colours. *Philosophical Magazine Series 4*, 7(45):254–264, 1854.

R. Gribonval, R. Jenatton, F. Bach, M. Kleinsteuber, and M. Seibert. Sample complexity of dictionary learning and other matrix factorizations. *preprint arXiv:1312.3790*, 2013.

References XII

P. Hall, G. Kerkyacharian, and D. Picard. On the minimax optimality of block thresholded wavelet estimators. *Statistica Sinica*, 9(1):33–49, 1999.

K. K. Herrity, A. C. Gilbert, and J. A. Tropp. Sparse approximation via iterative thresholding. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing, (ICASSP)*, 2006.

R. R. Hocking. A Biometrics invited paper. The analysis and selection of variables in linear regression. *Biometrics*, 32:1–49, 1976.

H. Hoefling. A path algorithm for the fused lasso signal approximator. *Journal of Computational and Graphical Statistics*, 19(4):984–1006, 2010.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

References XIII

J. Huang, Z. Zhang, and D. Metaxas. Learning with structured sparsity. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195 (1):215–243, 1968.

A. Hyvärinen, P. O. Hoyer, and M. Inki. Topographic independent component analysis. *Neural computation*, 13(7):1527–1558, 2001.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent component analysis*. John Wiley and Sons, 2004.

A. Hyvärinen, J. Hurri, and P. O. Hoyer. *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision*. Springer, 2009.

L. Jacob, G. Obozinski, and J.-P. Vert. Group Lasso with overlaps and graph Lasso. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

References XIV

R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010.

R. Jenatton, J-Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. *Journal of Machine Learning Research*, 12:2777–2824, 2011a.

R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for hierarchical sparse coding. *Journal of Machine Learning Research*, 12:2297–2334, 2011b.

K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun. Learning invariant features through topographic filter maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

References XV

K. Kavukcuoglu, M. Ranzato, and Y. LeCun. Fast inference in sparse coding algorithms with applications to object recognition. *preprint arXiv:1010.3467*, 2010.

K. Lange, D. R. Hunter, and I. Yang. Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, 9(1):1–20, 2000.

E. Le Pennec and S. Mallat. Sparse geometric image representations with bandelets. *IEEE Transactions on Image Processing*, 14(4): 423–438, 2005.

Y. LeCun, L. Bottou, G. Orr, and K. Muller. Efficient backprop. In G. Orr and Muller K., editors, *Neural Networks: Tricks of the trade*. Springer, 1998.

H. Lee, A. Battle, R. Raina, and A.Y. Ng. Efficient sparse coding algorithms. *Advances in Neural Information Processing Systems (NIPS)*, 19:801, 2007.

References XVI

J. Mairal and B. Yu. Complexity analysis of the Lasso regularization path. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2012.

J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*, 17(1):53–69, 2008a.

J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *SIAM Multiscale Modeling and Simulation*, 7(1):214–241, 2008b.

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009a.

J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009b.

References XVII

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11:19–60, 2010a.

J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Network flow algorithms for structured sparsity. In *Advances in Neural Information Processing Systems (NIPS)*, 2010b.

J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Convex and network flow optimization for structured sparsity. *Journal of Machine Learning Research*, 12:2681–2720, 2011.

J. Mairal, F. Bach, and J. Ponce. Task-driven dictionary learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4): 791–804, 2012.

S. Mallat. *A wavelet tour of signal processing*. Academic press, 2008. 3rd edition.

References XVIII

S. Mallat and Z. Zhang. Matching pursuit in a time-frequency dictionary. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

C. L. Mallows. Choosing variables in a linear regression: A graphical aid. unpublished paper presented at the Central Regional Meeting of the Institute of Mathematical Statistics, Manhattan, Kansas, 1964.

C. L. Mallows. Choosing a subset regression. unpublished paper presented at the Joint Statistical Meeting, Los Angeles, California, 1966.

H. Markowitz. Portfolio selection. *Journal of Finance*, 7(1):77–91, 1952.

J. C. Maxwell. On the theory of compound colours, and the relations of the colours of the spectrum. *Philosophical Transactions of the Royal Society of London*, pages 57–84, 1860.

C. A. Micchelli, J. M. Morales, and M. Pontil. Regularizers for structured sparsity. *Advances in Computational Mathematics*, 38(3): 455–489, 2013.

References XIX

N. M. Nasrabadi and R. A. King. Image coding using vector quantization: A review. *IEEE Transactions on Communications*, 36 (8):957–971, 1988.

B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24:227–234, 1995.

J. Nathans, D. Thomas, and D. S. Hogness. Molecular genetics of human color vision: the genes encoding blue, green, and red pigments. *Science*, 232(4747):193–202, 1986.

Y. Nesterov. *Introductory lectures on convex optimization: a basic course*. Kluwer Academic Publishers, 2004.

Y. Nesterov. Gradient methods for minimizing composite objective function. *Mathematical Programming*, 140(1):125–161, 2013.

I. Newton. Hypothesis explaining the properties of light. In *The History of the Royal Society*, volume 3, pages 247–269. T. Birch, 1675. text published in 1757.

References XX

R. D. Nowak and M. A. T. Figueiredo. Fast wavelet-based image deconvolution using the EM algorithm. In *Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers.*, 2001.

B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381: 607–609, 1996.

B. A. Olshausen and D. J. Field. How close are we to understanding V1? *Neural computation*, 17(8):1665–1699, 2005.

M. R. Osborne, B. Presnell, and B. A. Turlach. On the Lasso and its dual. *Journal of Computational and Graphical Statistics*, 9(2):319–37, 2000a.

M. R. Osborne, B. Presnell, and B. A. Turlach. A new approach to variable selection in least squares problems. *IMA journal of numerical analysis*, 20(3):389–403, 2000b.

References XXI

P. Paatero and U. Tapper. Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proceedings of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, 1993.

J. Pearl. On coding and filtering stationary signals by discrete fourier transforms (corresp.). *IEEE Transactions on Information Theory*, 19 (2):229–232, 1973.

D.-T. Pham. Fast algorithms for mutual information based independent component analysis. *IEEE Transactions on Signal Processing*, 52(10): 2690–2700, 2004.

W. Pratt. Spatial transform coding of color images. *IEEE Transactions on Communication Technology*, 19(6):980–992, 1971.

References XXII

M. Protter and M. Elad. Image sequence denoising via sparse and redundant representations. *IEEE Transactions on Image Processing*, 18(1):27–35, 2009.

F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, and J.P. Vert. Classification of microarray data using gene networks. *BMC bioinformatics*, 8(1):35, 2007.

R. Rigamonti, A. Sironi, V. Lepetit, and P. Fua. Learning separable filters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5): 465–471, 1978.

K. Ritter. Ein verfahren zur lösung parameterabhängiger, nichtlinearer maximum-probleme. *Mathematical Methods of Operations Research*, 6(4):149–166, 1962.

References XXIII

L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4): 259–268, 1992.

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

J. M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12): 3445–3462, 1993.

G. Sharma and H. J. Trussell. Digital color imaging. *IEEE Transactions on Image Processing*, 6(7):901–932, 1997.

E. P. Simoncelli and B. A. Olshausen. Natural image statistics and neural representation. *Annual review of neuroscience*, 24(1): 1193–1216, 2001.

References XXIV

E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multiscale transforms. *IEEE Transactions on Information Theory*, 38(2):587–607, 1992.

N. Srebro, J.D.M. Rennie, and T.S. Jaakkola. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, 2005.

J.-L. Starck, D. L. Donoho, and E. J. Candès. Astronomical image representation by the curvelet transform. *Astronomy and Astrophysics*, 398(2):785–800, 2003.

H. L. Taylor, S. C. Banks, and J. F. McCoy. Deconvolution with the ℓ_1 norm. *Geophysics*, 44(1):39–52, 1979.

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society Series B*, 58(1):267–288, 1996.

References XXV

R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused Lasso. *Journal of the Royal Statistical Society Series B*, 67(1):91–108, 2005.

B. A. Turlach, W. N. Venables, and S. J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005.

D. Vainsencher, S. Mannor, and A. M. Bruckstein. The sample complexity of dictionary learning. *Journal of Machine Learning Research*, 12:3259–3281, 2011.

H. von Helmholtz. LXXXI. on the theory of compound colours. *Philosophical Magazine Series 4*, 4(28):519–534, 1852.

M.J. Wainwright. Sharp thresholds for noisy and high-dimensional recovery of sparsity using ℓ_1-constrained quadratic programming. *IEEE Transactions on Information Theory*, 55(5):2183–2202, 2009.

References XXVI

D. Wrinch and H. Jeffreys. XLII. On certain fundamental principles of scientific inquiry. *Philosophical Magazine Series 6*, 42(249):369–390, 1921.

J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19 (11):2861–2873, 2010.

J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang. Coupled dictionary training for image super-resolution. *IEEE Transactions on Image Processing*, 21(8):3467–3478, 2012.

T. Young. *A course of lectures on natural philosophy and the mechanical art*. Taylor and Watson, 1845.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B*, 68:49–67, 2006.

References XXVII

M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse representations. In *Curves and Surfaces*, pages 711–730. Springer, 2012.

P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.

P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *Annals of Statistics*, 37 (6A):3468–3497, 2009.

S. C. Zhu and D. Mumford. Prior learning and gibbs reaction-diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19 (11):1236–1250, 1997.

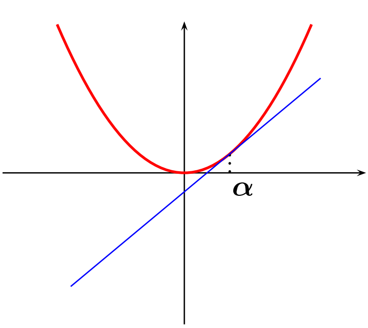
References XXVIII

S.-C. Zhu, C.-E. Guo, Y. Wang, and Z. Xu. What are textons? *International Journal of Computer Vision*, 62(1-2):121–143, 2005.

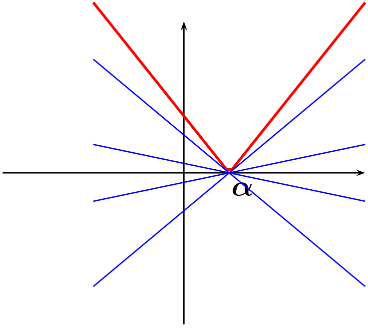
H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67(2): 301–320, 2005.

Appendix

Basic convex optimization tools: subgradients



(a) Smooth case



(b) Non-smooth case

Figure: Gradients and subgradients for smooth and non-smooth functions.

$$\partial f(\boldsymbol{\alpha}) \triangleq \{\boldsymbol{\kappa} \in \mathbb{R}^p \mid f(\boldsymbol{\alpha}) + \boldsymbol{\kappa}^\top(\boldsymbol{\alpha}' - \boldsymbol{\alpha}) \leq f(\boldsymbol{\alpha}') \text{ for all } \boldsymbol{\alpha}' \in \mathbb{R}^p\}.$$

Basic convex optimization tools: subgradients

Some nice properties

- $\partial f(\boldsymbol{\alpha}) = \{g\}$ iff f differentiable at $\boldsymbol{\alpha}$ and $g = \nabla f(\boldsymbol{\alpha})$.
- many calculus rules: $\partial(\gamma f + \mu g) = \gamma \partial f + \mu \partial g$ for $\gamma, \mu > 0$.

for more details, see Boyd and Vandenberghe [2004], Bertsekas [1999], Borwein and Lewis [2006] and S. Boyd's course at Stanford.

Optimality conditions

For $g : \mathbb{R}^p \to \mathbb{R}$ convex,

- g differentiable: $\boldsymbol{\alpha}^\star$ minimizes g iff $\nabla g(\boldsymbol{\alpha}^\star) = 0$.
- g nondifferentiable: $\boldsymbol{\alpha}^\star$ minimizes g iff $0 \in \partial g(\boldsymbol{\alpha}^\star)$.

Careful: the concept of subgradient requires a function to be above its tangents. It does only make sense for convex functions!

Basic convex optimization tools: dual-norm

Definition

Let $\boldsymbol{\kappa}$ be in $\mathbb{R}^p$,

$$\|\boldsymbol{\kappa}\|_* \triangleq \max_{\boldsymbol{\alpha} \in \mathbb{R}^p : \|\boldsymbol{\alpha}\| \leq 1} \boldsymbol{\alpha}^\top \boldsymbol{\kappa}.$$

Exercises

- $\|\boldsymbol{\alpha}\|_{**} = \|\boldsymbol{\alpha}\|$ (true in finite dimension)
- ℓ_2 is dual to itself.
- ℓ_1 and ℓ_∞ are dual to each other.
- ℓ_q and ℓ'_q are dual to each other if $\frac{1}{q} + \frac{1}{q'} = 1$.
- similar relations for spectral norms on matrices.
- $\partial \|\boldsymbol{\alpha}\| = \{\boldsymbol{\kappa} \in \mathbb{R}^p \;\text{ s.t. }\; \|\boldsymbol{\kappa}\|_* \leq 1 \text{ and } \boldsymbol{\kappa}^\top \boldsymbol{\alpha} = \|\boldsymbol{\alpha}\|\}$.

Optimality conditions

Let $f : \mathbb{R}^p \rightarrow \mathbb{R}$ be convex differentiable and $\|.\|$ be any norm.

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} f(\boldsymbol{\alpha}) + \lambda \|\boldsymbol{\alpha}\|.$$

$\boldsymbol{\alpha}$ is solution if and only if

$$0 \in \partial(f(\boldsymbol{\alpha}) + \lambda\|\boldsymbol{\alpha}\|) = \nabla f(\boldsymbol{\alpha}) + \lambda \partial \|\boldsymbol{\alpha}\|$$

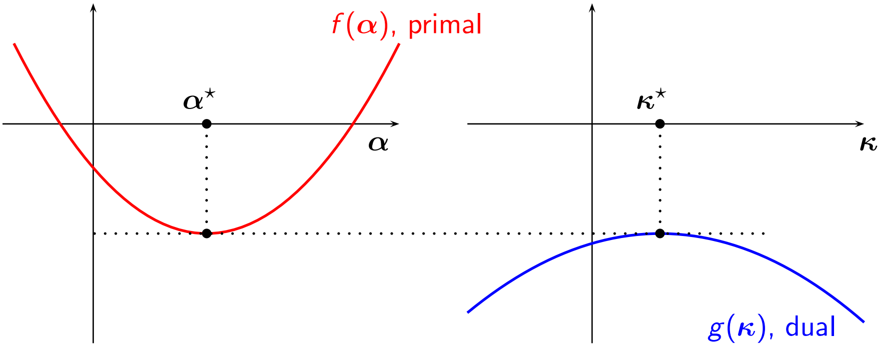
Since $\partial\|\boldsymbol{\alpha}\| = \{\boldsymbol{\kappa} \in \mathbb{R}^p \ \text{ s.t. } \ \|\boldsymbol{\kappa}\|_* \leq 1 \text{ and } \boldsymbol{\kappa}^\top \boldsymbol{\alpha} = \|\boldsymbol{\alpha}\|\}$,

General optimality conditions:

$$\|\nabla f(\boldsymbol{\alpha})\|_* \leq \lambda \ \text{ and } \ -\nabla f(\boldsymbol{\alpha})^\top \boldsymbol{\alpha} = \lambda \|\boldsymbol{\alpha}\|.$$

Convex Duality

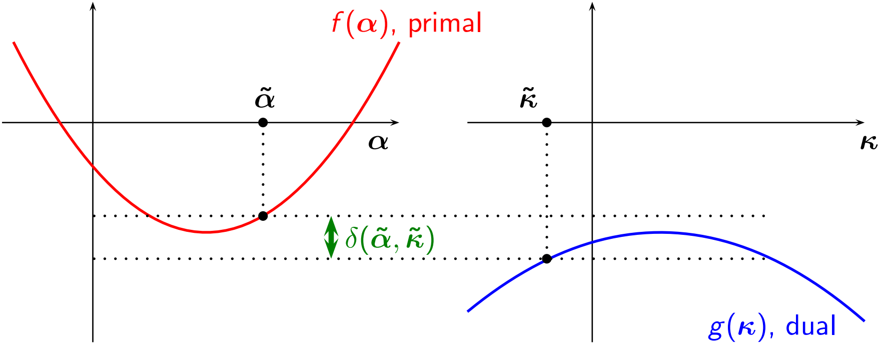
Strong Duality



Strong duality means that $\max_{\boldsymbol{\kappa}} g(\boldsymbol{\kappa}) = \min_{\boldsymbol{\alpha}} f(\boldsymbol{\alpha})$

Convex Duality

Duality Gaps



Strong duality means that $\max_{\boldsymbol{\kappa}} g(\boldsymbol{\kappa}) = \min_{\boldsymbol{\alpha}} f(\boldsymbol{\alpha})$

The duality gap guarantees us that $0 \leq f(\tilde{\boldsymbol{\alpha}}) - f(\boldsymbol{\alpha}^\star) \leq \delta(\tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\kappa}})$.