Sparse Coding and Dictionary Learning for Image Analysis

Julien Mairal

Willow group - INRIA - ENS - Paris

ERMITES 2010, September 2010

ERMITES 2010

At Hyères, September 28 - 30, 2010

Acknowledgements

Francis
Bach

Jean
Ponce

Rodolphe
Jenatton

Guillaume
Obozinski

Guillermo
Sapiro

Florent
Couzinié-Devy

Y-Lan
Boureau

What this lecture is about?

- **Why sparsity, what for and how?**
- **Signal and image processing**: Restoration, reconstruction.
- **Machine learning**: Selecting relevant features.
- **Computer vision**: Modelling the local appearance of image patches.
- **Computer vision**: Recent (and intriguing) results in bags of words models.
- **Optimization**: Solving challenging problems.

1. Image Processing Applications
   - Image Denoising
   - Inpainting, Demosaicking
   - Video Processing
   - Other Applications
2. Sparse Linear Models and Dictionary Learning
3. Computer Vision Applications
4. Optimization for sparse methods

The Image Denoising Problem

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

Sparse representations for image restoration

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

Energy minimization problem - MAP estimation

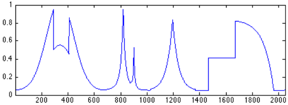
$$E(\mathbf{x}) = \underbrace{\frac{1}{2}\|\mathbf{y}-\mathbf{x}\|_2^2}_{\text{relation to measurements}} + \underbrace{Pr(\mathbf{x})}_{\text{image model (-log prior)}}$$

Some classical priors

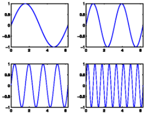
- Smoothness $\lambda\|\mathcal{L}\mathbf{x}\|_2^2$
- Total variation $\lambda\|\nabla\mathbf{x}\|_1^2$
- MRF priors
- ...

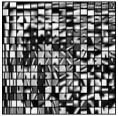
What is a Sparse Linear Model?

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



Let $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_p] \in \mathbb{R}^{m \times p}$ be a set of normalized "basis vectors".
We call it **dictionary**.





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

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

First Important Idea

Why Sparsity?

A dictionary can be good for representing a class of signals, but not for representing white Gaussian noise.

The Sparse Decomposition Problem

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \underbrace{\frac{1}{2}\|\mathbf{y}-\mathbf{D}\boldsymbol{\alpha}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda\psi(\boldsymbol{\alpha})}_{\text{sparsity-inducing regularization}}$$

ψ induces sparsity in $\boldsymbol{\alpha}$. It can be

- the ℓ_0 "pseudo-norm". $\|\boldsymbol{\alpha}\|_0 \triangleq \#\{i \text{ s.t. } \boldsymbol{\alpha}[i] \neq 0\}$ (NP-hard)
- the ℓ_1 norm. $\|\boldsymbol{\alpha}\|_1 \triangleq \sum_{i=1}^{p} |\boldsymbol{\alpha}[i]|$ (convex),
- ...

This is a selection problem. When ψ is the ℓ_1-norm, the problem is called Lasso [Tibshirani, 1996] or basis pursuit [Chen et al., 1999]

Sparse representations for image restoration

Designed dictionaries

[Haar, 1910], [Zweig, Morlet, Grossman ∼70s], [Meyer, Mallat, Daubechies, Coifman, Donoho, Candes ∼80s-today]. . . (see [Mallat, 1999])

Wavelets, Curvelets, Wedgelets, Bandlets, . . . lets

Learned dictionaries of patches

[Olshausen and Field, 1997], [Engan et al., 1999], [Lewicki and Sejnowski, 2000], [Aharon et al., 2006] , [Roth and Black, 2005], [Lee et al., 2007]

$$\min_{\boldsymbol{\alpha}_i, \mathbf{D} \in \mathcal{C}} \sum_i \underbrace{\frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2}_{\text{reconstruction}} + \underbrace{\lambda \psi(\boldsymbol{\alpha}_i)}_{\text{sparsity}}$$

- $\psi(\boldsymbol{\alpha}) = \|\boldsymbol{\alpha}\|_0$ (“ℓ_0 pseudo-norm”)
- $\psi(\boldsymbol{\alpha}) = \|\boldsymbol{\alpha}\|_1$ (ℓ_1 norm)

Solving the denoising problem

[Elad and Aharon, 2006]

- Extract all overlapping 8×8 patches $\mathbf{y}_i$.
- Solve a matrix factorization problem:

$$\min_{\boldsymbol{\alpha}_i, \mathbf{D} \in \mathcal{C}} \sum_{i=1}^{n} \underbrace{\frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2}_{\text{reconstruction}} + \underbrace{\lambda\psi(\boldsymbol{\alpha}_i)}_{\text{sparsity}},$$

  with $n > 100,000$

- Average the reconstruction of each patch.

Sparse representations for image restoration

K-SVD: [Elad and Aharon, 2006]

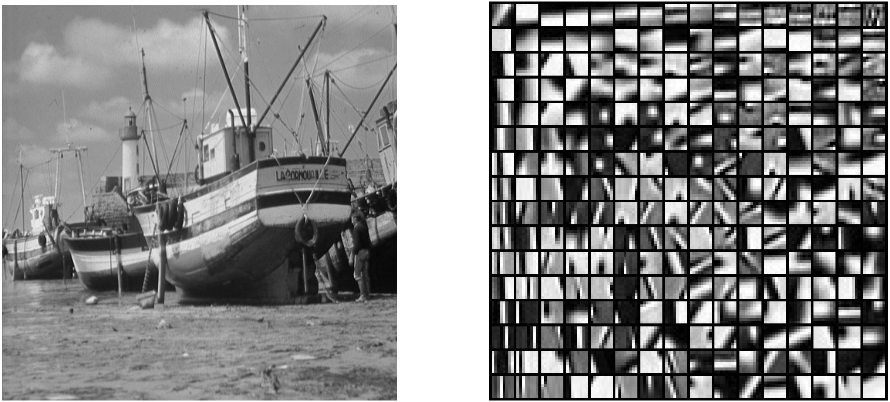
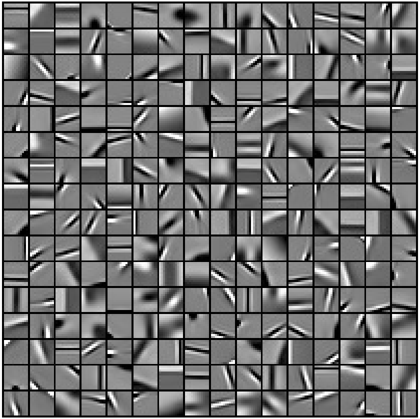
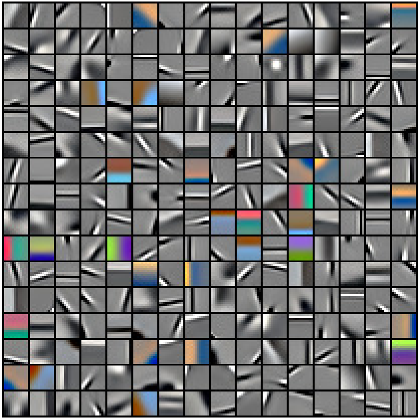


Figure: Dictionary trained on a noisy version of the image boat.

Sparse representations for image restoration

Grayscale vs color image patches



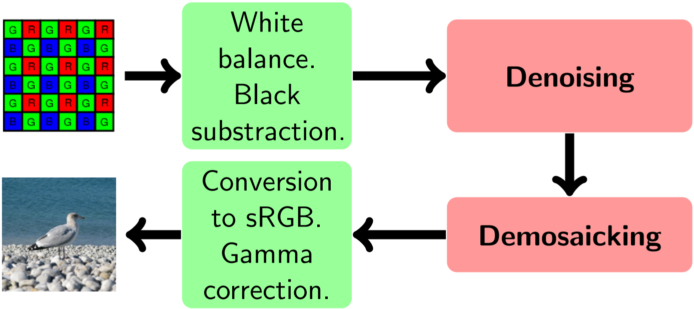


Sparse representations for image restoration

Inpainting, Demosaicking

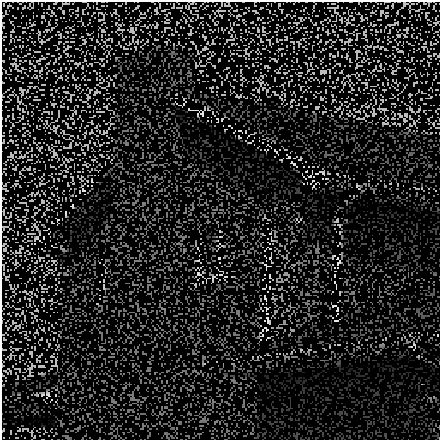
$$\min_{\mathbf{D}\in\mathcal{C},\boldsymbol{\alpha}} \sum_i \frac{1}{2}\|\boldsymbol{\beta}_i(\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i)\|_2^2 + \lambda_i\psi(\boldsymbol{\alpha}_i)$$

RAW Image Processing



Sparse representations for image restoration

[Mairal, Sapiro, and Elad, 2008d]



Sparse representations for image restoration

Inpainting, [Mairal, Elad, and Sapiro, 2008b]

Sparse representations for image restoration

Inpainting, [Mairal, Elad, and Sapiro, 2008b]

Sparse representations for video restoration

Key ideas for video processing

[Protter and Elad, 2009]

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

Sparse representations for image restoration
Inpainting, [Mairal, Sapiro, and Elad, 2008d]

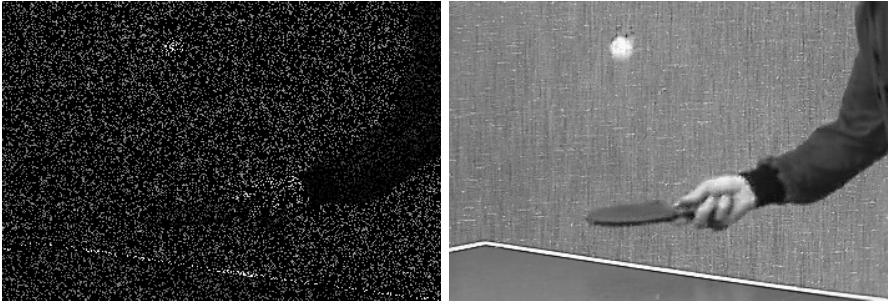


Figure: Inpainting results.

Sparse representations for image restoration
Inpainting, [Mairal, Sapiro, and Elad, 2008d]

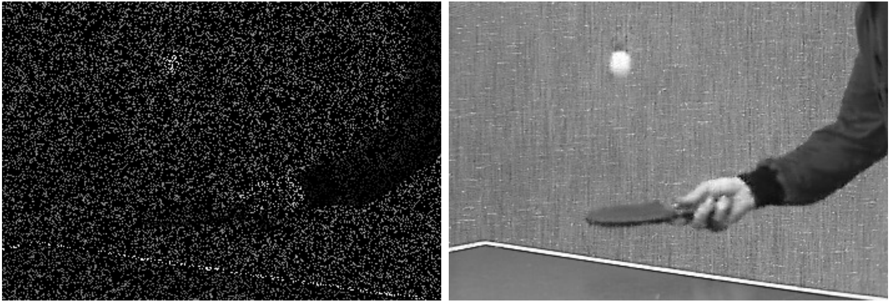


Figure: Inpainting results.

Sparse representations for image restoration

Inpainting, [Mairal, Sapiro, and Elad, 2008d]

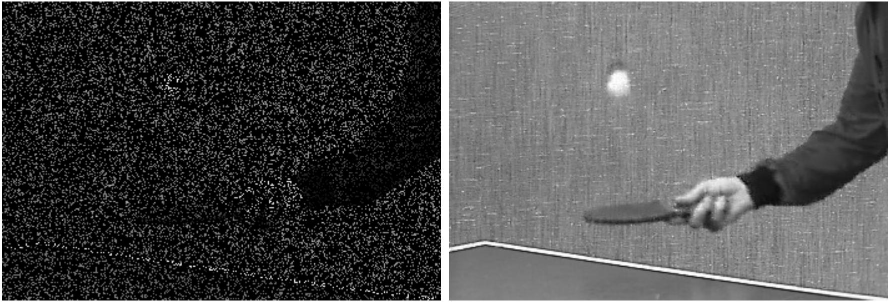


Figure: Inpainting results.

Sparse representations for image restoration

Inpainting, [Mairal, Sapiro, and Elad, 2008d]

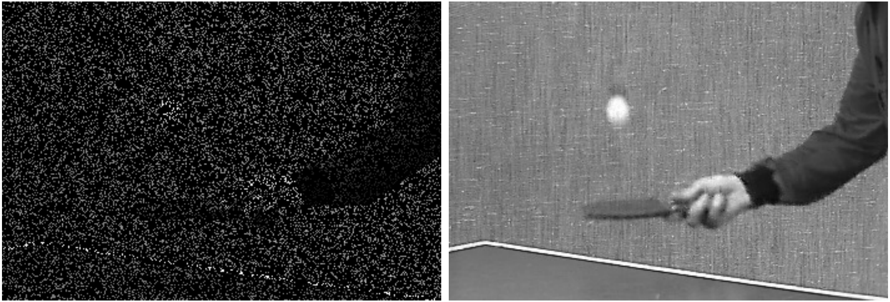


Figure: Inpainting results.

Sparse representations for image restoration
Inpainting, [Mairal, Sapiro, and Elad, 2008d]

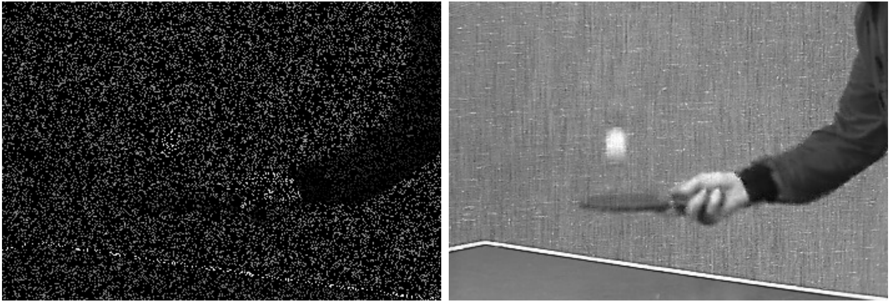


Figure: Inpainting results.

Sparse representations for image restoration

Color video denoising, [Mairal, Sapiro, and Elad, 2008d]

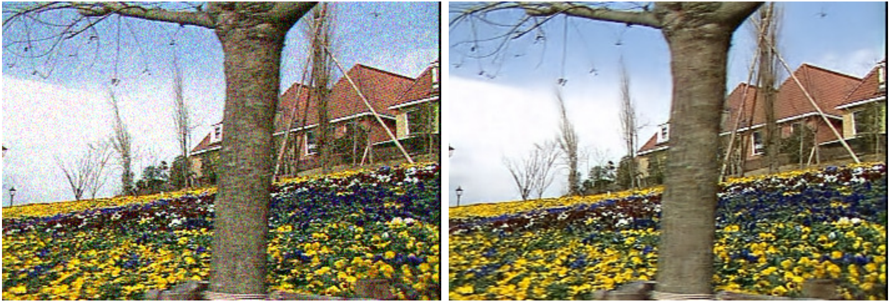


Figure: Denoising results. $\sigma = 25$

Sparse representations for image restoration
Color video denoising, [Mairal, Sapiro, and Elad, 2008d]

Figure: Denoising results. $\sigma = 25$

Sparse representations for image restoration

Color video denoising, [Mairal, Sapiro, and Elad, 2008d]

Figure: Denoising results. $\sigma = 25$

Sparse representations for image restoration

Color video denoising, [Mairal, Sapiro, and Elad, 2008d]

Figure: Denoising results. $\sigma = 25$

Sparse representations for image restoration

Color video denoising, [Mairal, Sapiro, and Elad, 2008d]

Figure: Denoising results. $\sigma = 25$

Digital Zooming

Couzinie-Devy, 2010, Original

Digital Zooming

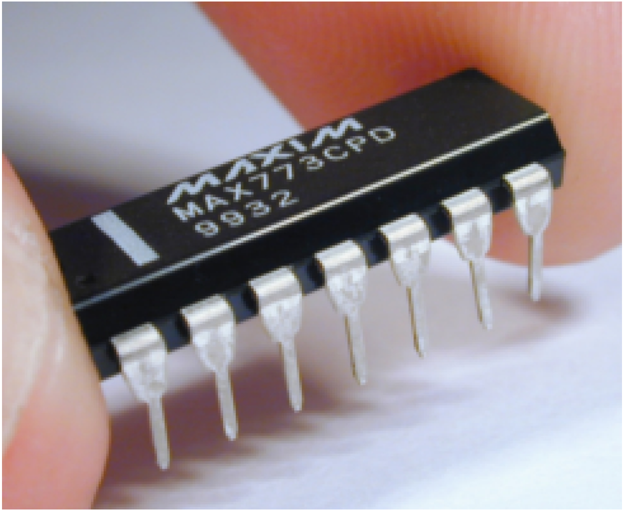
Couzinie-Devy, 2010, Bicubic

Digital Zooming

Couzinie-Devy, 2010, Proposed method

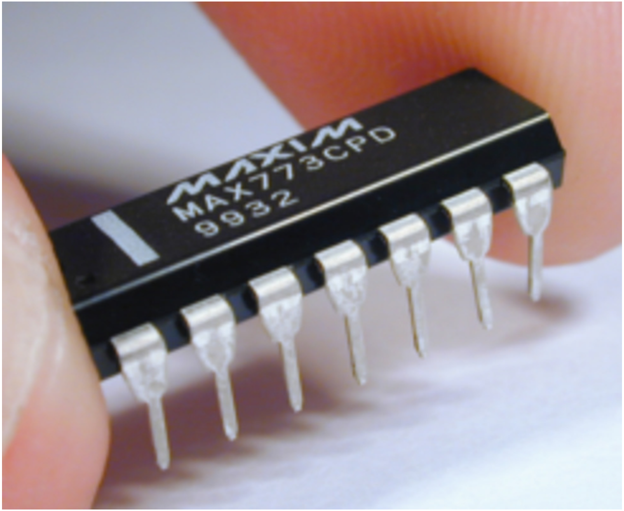
Digital Zooming

Couzinie-Devy, 2010, Original



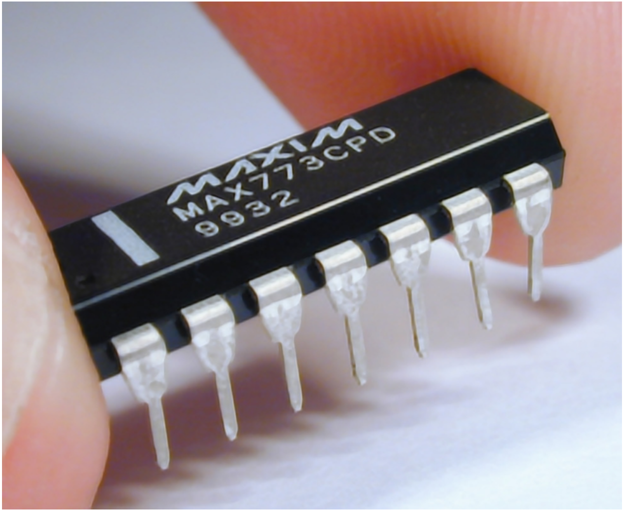
Digital Zooming

Couzinie-Devy, 2010, Bicubic



Digital Zooming

Couzinie-Devy, 2010, Proposed approach



Inverse half-toning

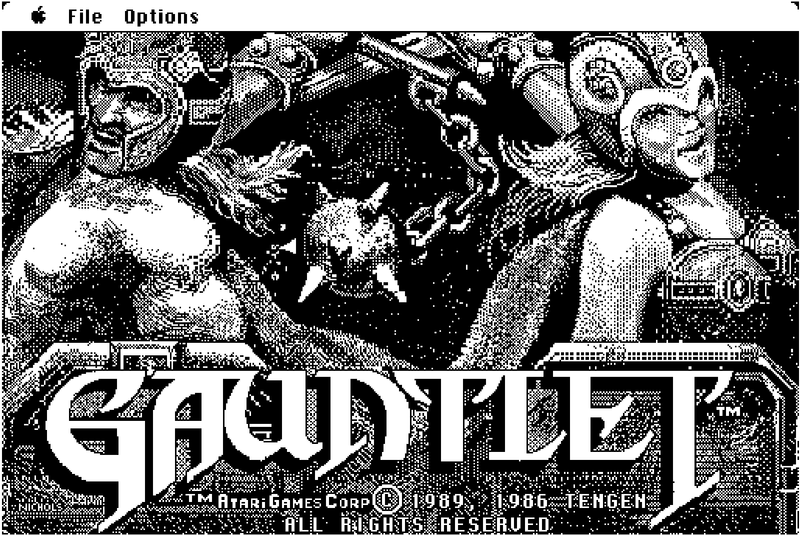
Original

Inverse half-toning

Reconstructed image

Inverse half-toning

Original



Inverse half-toning

Reconstructed image

Inverse half-toning

Original

Inverse half-toning

Reconstructed image

Inverse half-toning

Original

Inverse half-toning

Reconstructed image

Inverse half-toning

Original

Inverse half-toning

Reconstructed image

One short slide on compressed sensing

Important message

Sparse coding is not "compressed sensing".

Compressed sensing is a theory [see Candes, 2006] saying that a sparse signal can be recovered from a few linear measurements under some conditions.

- Signal Acquisition: $\mathbf{W}^\top \mathbf{y}$, where $\mathbf{W} \in \mathbb{R}^{m \times s}$ is a "sensing" matrix with $s \ll m$.
- Signal Decoding: $\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \|\boldsymbol{\alpha}\|_1$ s.t. $\mathbf{W}^\top \mathbf{y} = \mathbf{W}^\top \mathbf{D} \boldsymbol{\alpha}$.

with extensions to approximately sparse signals, noisy measurements.

Remark

The dictionaries we are using in this lecture do not satisfy the recovery assumptions of compressed sensing.

Important messages

- Patch-based approaches are achieving state-of-the-art results for many image processing task.
- Dictionary Learning adapts to the data you want to restore.
- Dictionary Learning is well adapted to data that admit sparse representation. **Sparsity is for sparse data only**.

Next topics

- A bit of machine learning.
- Why does the ℓ_1-norm induce sparsity?
- Some properties of the Lasso.
- Links between dictionary learning and matrix factorization techniques.
- A simple algorithm for learning dictionaries.
- Beyond sparsity: Group-sparsity, Structured Sparsity

Sparse Linear Model: Machine Learning 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.
- $\{1, \ldots, N\}$ for **multiclass** classification problems.
- $\mathbb{R}$ for **regression** problems.

In a linear model, on assumes a relation $y \approx \mathbf{w}^\top \mathbf{x}$ (or $y \approx \text{sign}(\mathbf{w}^\top \mathbf{x})$), and solves

$$\min_{\mathbf{w} \in \mathbb{R}^p} \underbrace{\frac{1}{n} \sum_{i=1}^{n} \ell(y^i, \mathbf{w}^\top \mathbf{x}^i)}_{\text{data-fitting}} + \underbrace{\lambda \Omega(\mathbf{w})}_{\text{regularization}} .$$

Sparse Linear Models: Machine Learning Point of View

A few examples:

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

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

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

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

The purpose of the regularization is to add **additional a-priori knowledge** in the regularization.

Sparse Linear Models: the Lasso

- Signal processing: $\mathbf{D}$ is a dictionary in $\mathbb{R}^{n \times p}$,

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

- Machine Learning:

$$\min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{2n}\sum_{i=1}^{n}(y^i - \mathbf{x}^{i\top}\mathbf{w})^2 + \lambda\|\mathbf{w}\|_1 = \min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{2n}\|\mathbf{y} - \mathbf{X}^\top\mathbf{w}\|_2^2 + \lambda\|\mathbf{w}\|_1,$$

with $\mathbf{X} \triangleq [\mathbf{x}^1, \ldots, \mathbf{x}^n]$, and $\mathbf{y} \triangleq [y^1, \ldots, y^n]^\top$.

Useful tool in signal processing, machine learning, statistics,... as long as one wishes to **select** features.

Why does the ℓ_1-norm induce sparsity?

Exemple: quadratic problem in 1D

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

Piecewise quadratic function with a kink at zero.

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

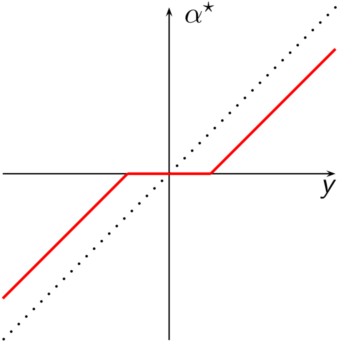
Optimality conditions. α is optimal iff:

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

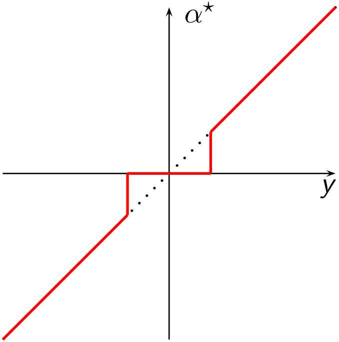
The solution is a **soft-thresholding**:

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

Why does the ℓ_1-norm induce sparsity?



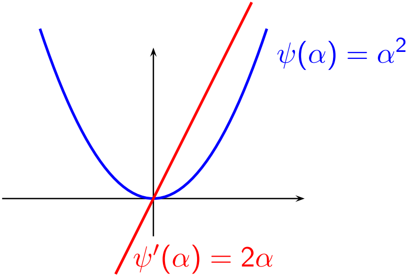
(a) soft-thresholding operator

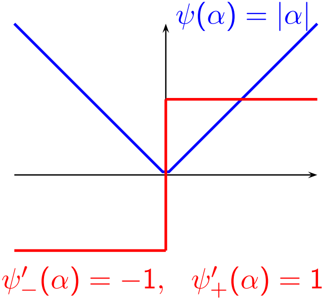


(b) hard-thresholding operator

Why does the ℓ_1-norm induce sparsity?

Analysis of the norms in 1D

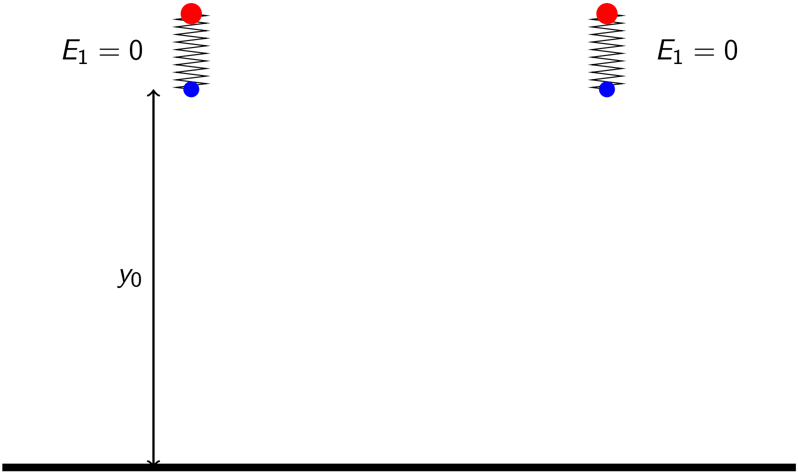




The gradient of the ℓ_2-norm vanishes when α get close to 0. On its differentiable part, the norm of the gradient of the ℓ_1-norm is constant.

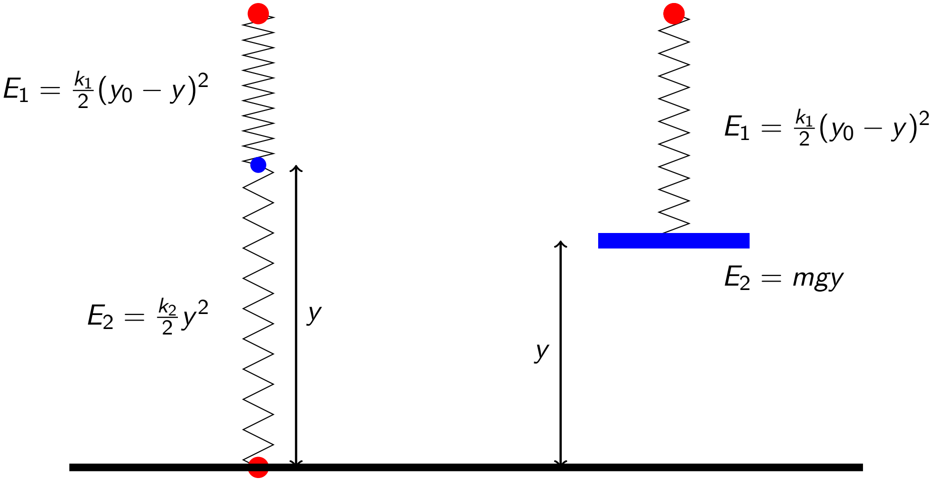
Why does the ℓ_1-norm induce sparsity?

Physical illustration



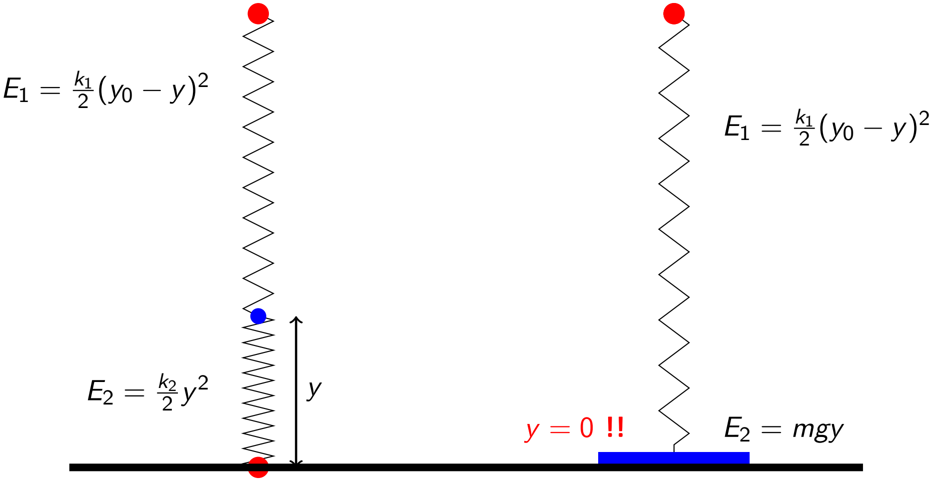
Why does the ℓ_1-norm induce sparsity?

Physical illustration



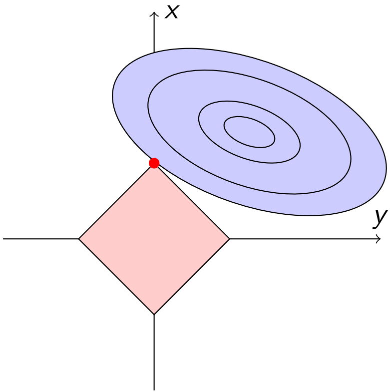
Why does the ℓ_1-norm induce sparsity?

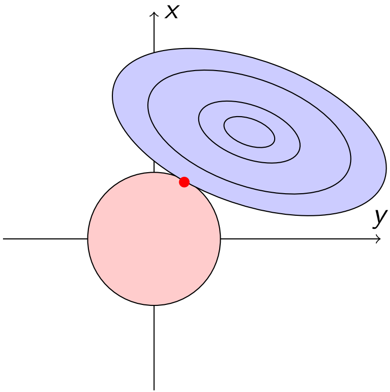
Physical illustration



Why does the ℓ_1-norm induce sparsity?

Geometric explanation





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

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

Important property of the Lasso

Piecewise linearity of the regularization path

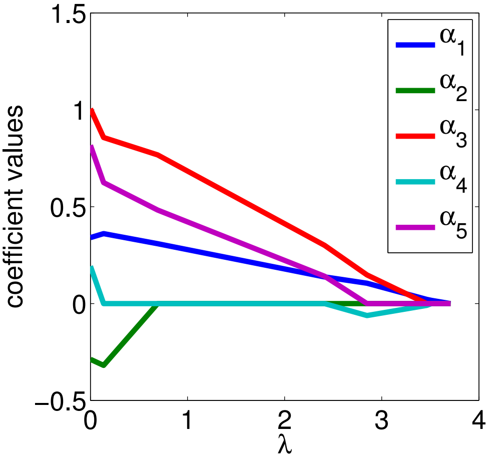


Figure: Regularization path of the Lasso

Optimization for Dictionary Learning

$$\min_{\substack{\boldsymbol{\alpha}\in\mathbb{R}^{p\times n}\\ \mathbf{D}\in\mathcal{C}}} \sum_{i=1}^{n} \frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1$$

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

- Classical optimization alternates between $\mathbf{D}$ and $\boldsymbol{\alpha}$.
- Good results, but slow!
- Instead use online learning [Mairal et al., 2009a]

Optimization for Dictionary Learning
Inpainting a 12-Mpixel photograph

Optimization for Dictionary Learning
Inpainting a 12-Mpixel photograph

Optimization for Dictionary Learning
Inpainting a 12-Mpixel photograph

Optimization for Dictionary Learning
Inpainting a 12-Mpixel photograph

Matrix Factorization Problems and Dictionary Learning

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^{p \times n} \\ \mathbf{D} \in \mathcal{C}}} \sum_{i=1}^{n} \frac{1}{2} \|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1$$

can be rewritten

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^{p \times n} \\ \mathbf{D} \in \mathcal{C}}} \frac{1}{2} \|\mathbf{Y} - \mathbf{D}\boldsymbol{\alpha}\|_F^2 + \lambda \|\boldsymbol{\alpha}\|_1,$$

where $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]$ and $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_n]$.

Matrix Factorization Problems and Dictionary Learning

PCA

$$\min_{\substack{\boldsymbol{\alpha}\in\mathbb{R}^{p\times n}\\ \mathbf{D}\in\mathbb{R}^{m\times p}}} \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\boldsymbol{\alpha}\|_F^2 \;\; \text{s.t.} \;\; \mathbf{D}^\top\mathbf{D}=\mathbf{I} \text{ and } \boldsymbol{\alpha}\boldsymbol{\alpha}^\top \text{ is diagonal.}$$

$\mathbf{D}=[\mathbf{d}_1,\ldots,\mathbf{d}_p]$ are the principal components.

Matrix Factorization Problems and Dictionary Learning

Hard clustering

$$\min_{\substack{\boldsymbol{\alpha}\in\{0,1\}^{p\times n} \\ \mathbf{D}\in\mathbb{R}^{m\times p}}} \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\boldsymbol{\alpha}\|_F^2 \ \text{ s.t. } \ \forall i \in \{1,\ldots,p\}, \ \sum_{j=1}^{p} \boldsymbol{\alpha}_i[j] = 1.$$

$\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_p]$ are the centroids of the p clusters.

Matrix Factorization Problems and Dictionary Learning

Soft clustering

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}_+^{p \times n} \\ \mathbf{D} \in \mathbb{R}^{m \times p}}} \frac{1}{2} \|\mathbf{Y} - \mathbf{D}\boldsymbol{\alpha}\|_F^2, \quad \text{s.t. } \forall i \in \{1, \ldots, p\}, \ \sum_{j=1}^{p} \boldsymbol{\alpha}_i[j] = 1.$$

$\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_p]$ are the centroids of the p clusters.

Matrix Factorization Problems and Dictionary Learning

Non-negative matrix factorization [Lee and Seung, 2001]

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}_+^{p \times n} \\ \mathbf{D} \in \mathbb{R}_+^{m \times p}}} \frac{1}{2} \|\mathbf{Y} - \mathbf{D}\boldsymbol{\alpha}\|_F^2$$

Matrix Factorization Problems and Dictionary Learning

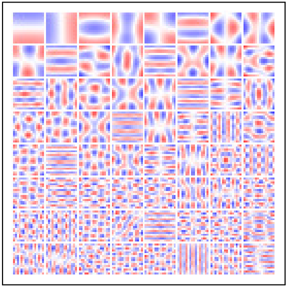
NMF+sparsity?

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}_+^{p \times n} \\ \mathbf{D} \in \mathbb{R}_+^{m \times p}}} \frac{1}{2} \|\mathbf{Y} - \mathbf{D}\boldsymbol{\alpha}\|_F^2 + \lambda \|\boldsymbol{\alpha}\|_1.$$

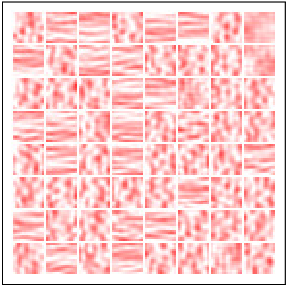
Most of these formulations can be addressed the same types of algorithms.

Matrix Factorization Problems and Dictionary Learning

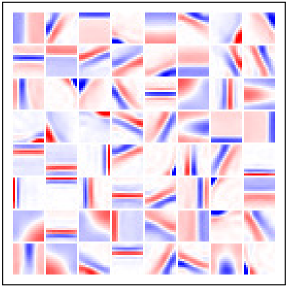
Natural Patches



(a) PCA



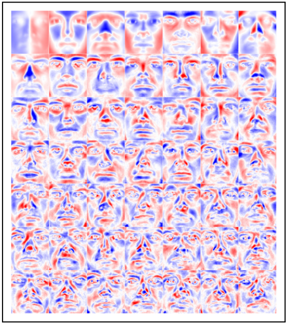
(b) NNMF



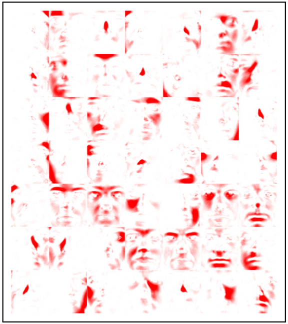
(c) DL

Matrix Factorization Problems and Dictionary Learning

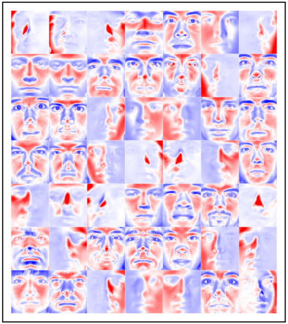
Faces



(d) PCA



(e) NNMF



(f) DL

Sparsity-Inducing Norms (1/2)

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \quad \overbrace{f(\boldsymbol{\alpha})}^{\text{data fitting term}} + \lambda \underbrace{\Omega(\boldsymbol{\alpha})}_{\text{sparsity-inducing norm}}$$

Standard approach to enforce sparsity in learning procedures:

- Regularizing by a **sparsity-inducing norm** ψ.
- The effect of Ω is to set some $\boldsymbol{\alpha}[j]$'s to zero, depending on the regularization parameter $\lambda \geq 0$.

The most popular choice for ψ:

- The ℓ_1 norm, $\|\boldsymbol{\alpha}\|_1 = \sum_{j=1}^{p} |\boldsymbol{\alpha}[j]|$.
- For the square loss, Lasso [Tibshirani, 1996].
- However, the ℓ_1 norm encodes poor information, just **cardinality**!

Sparsity-Inducing Norms (2/2)

Another popular choice for ψ:

- The ℓ_1-ℓ_2 norm,

$$\sum_{G \in \mathcal{G}} \|\boldsymbol{\alpha}_G\|_2 = \sum_{G \in \mathcal{G}} \Big(\sum_{j \in \mathcal{G}} \boldsymbol{\alpha}_j^2 \Big)^{1/2}, \text{ with } \mathcal{G} \text{ a } \textbf{partition} \text{ of } \{1, \ldots, p\}.$$

- The ℓ_1-ℓ_2 norm sets to zero **groups of non-overlapping variables** (as opposed to single variables for the ℓ_1 norm).
- For the square loss, group Lasso [Yuan and Lin, 2006].
- However, the ℓ_1-ℓ_2 norm encodes fixed/static prior information, requires to know in advance how to group the variables !

Applications:

- Selecting groups of features instead of individual variables.
- Multi-task learning, multiple kernel learning.

Non-local Sparse Image Models

Image Self-Similarities, [Buades et al., 2006, Efros and Leung, 1999, Dabov et al., 2007]

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

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

Non-local Sparse Image Models

Image Self-Similarities, [Buades et al., 2006, Efros and Leung, 1999, Dabov et al., 2007]

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

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

Successful application to texture synthesis: Efros and Leung [1999]
... to image denoising **(Non-Local Means)**: Buades et al. [2006]
... to image demosaicking: Buades et al. [2009]

Non-local Sparse Image Models

Image Self-Similarities, [Buades et al., 2006, Efros and Leung, 1999, Dabov et al., 2007]

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

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

Successful application to texture synthesis: Efros and Leung [1999]
...to image denoising **(Non-Local Means)**: Buades et al. [2006]
...to image demosaicking: Buades et al. [2009]

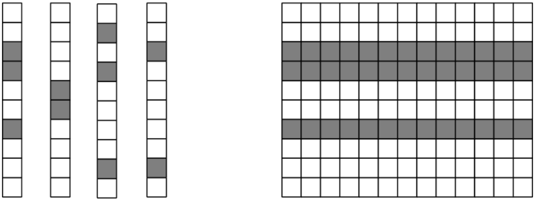
Block-Matching with 3D filtering (BM3D) Dabov et al. [2007], Similar patches are jointly denoised with orthogonal wavelet thresholding + several (good) heuristics: $\Longrightarrow$ state-of-the-art denoising results, less artefacts, higher PSNR.

Non-local Sparse Image Models

- **non-local means**: **stable** estimator. Can fail when there are no self-similarities.
- **sparse representations**: "unique" patches also admit a sparse approximation on the learned dictionary. potentially **unstable** decompositions.

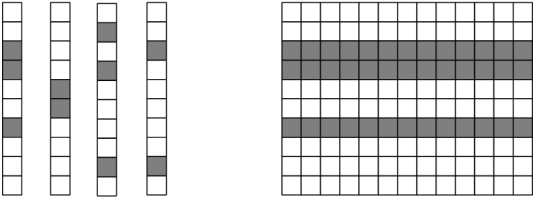
Improving the stability of sparse decompositions is a current topic of research in statistics Bach [2008], Meinshausen and Buehlmann [2010].

Mairal et al. [2009b]: Similar patches should admit similar patterns:



Sparsity vs. joint sparsity

Non-local Sparse Image Models



Sparsity vs. joint sparsity

Joint sparsity is achieved through specific regularizers such as

$$
\begin{aligned}
\|\mathbf{A}\|_{0,\infty} &\triangleq \sum_{i=1}^{p} \|\boldsymbol{\alpha}^i\|_0, \quad \text{(not convex, not a norm)} \\
\|\mathbf{A}\|_{1,2} &\triangleq \sum_{i=1}^{p} \|\boldsymbol{\alpha}^i\|_2. \quad \text{(convex norm)}
\end{aligned}
\tag{2}
$$

Non-local Sparse Image Models

Basic scheme for image denoising:

1. Cluster patches

$$S_i \triangleq \{j = 1, \ldots, n \ \text{ s.t. } \ \|\mathbf{y}_i - \mathbf{y}_j\|_2^2 \leq \xi\}, \tag{3}$$

2. Learn a dictionary with group-sparsity regularization

$$\min_{(\mathbf{A}_i)_{i=1}^n, \mathbf{D} \in \mathcal{C}} \sum_{i=1}^{n} \frac{\|\mathbf{A}_i\|_{1,2}}{|S_i|} \ \text{ s.t. } \ \forall i \sum_{j \in S_i} \|\mathbf{y}_j - \mathbf{D}\boldsymbol{\alpha}_{ij}\|_2^2 \leq \varepsilon_i \tag{4}$$

3. Estimate the final image by averaging the representations

Non-local Sparse Image Models

Basic scheme for image denoising:

1. Cluster patches

$$S_i \triangleq \{j = 1, \ldots, n \ \text{ s.t. } \ \|\mathbf{y}_i - \mathbf{y}_j\|_2^2 \leq \xi\}, \tag{3}$$

2. Learn a dictionary with group-sparsity regularization

$$\min_{(\mathbf{A}_i)_{i=1}^n, \mathbf{D} \in \mathcal{C}} \sum_{i=1}^n \frac{\|\mathbf{A}_i\|_{1,2}}{|S_i|} \ \text{ s.t. } \ \forall i \sum_{j \in S_i} \|\mathbf{y}_j - \mathbf{D}\boldsymbol{\alpha}_{ij}\|_2^2 \leq \varepsilon_i \tag{4}$$

3. Estimate the final image by averaging the representations

Details:

- Greedy clustering (linear time) and online learning.
- Eventually use two passes.
- Use non-convex regularization for the final reconstruction.

Non-local Sparse Image Models

Denoising results, synthetic noise

Average PSNR on 10 standard images (higher is better)

σ	GSM	FOE	KSVD	BM3D	SC	LSC	LSSC
5	37.05	37.03	37.42	37.62	37.46	37.66	**37.67**
10	33.34	33.11	33.62	34.00	33.76	33.98	**34.06**
15	31.31	30.99	31.58	32.05	31.72	31.99	**32.12**
20	29.91	29.62	30.18	30.73	30.29	30.60	**30.78**
25	28.84	28.36	29.10	29.72	29.18	29.52	**29.74**
50	25.66	24.36	25.61	26.38	25.83	26.18	**26.57**
100	22.80	21.36	22.10	23.25	22.46	22.62	**23.39**

Improvement over BM3D is significant only for large values of σ.
The comparison is made with GSM (Gaussian Scale Mixture) Portilla et al. [2003], FOE (Field of Experts) Roth and Black [2005], KSVD Elad and Aharon [2006] and BM3D Dabov et al. [2007].

Non-local Sparse Image Models

Denoising results, synthetic noise

Non-local Sparse Image Models

Denoising results, synthetic noise

Non-local Sparse Image Models

Demosaicking results, Kodak database

Average PSNR on the Kodak dataset (24 images)

Im.	AP	DL	LPA	SC	LSC	LSSC
Av.	39.21	40.05	40.52	40.88	41.13	**41.39**

The comparison is made with AP (Alternative Projections) Gunturk et al. [2002], DL Zhang and Wu [2005] and LPA Paliy et al. [2007] (best known result on this database).

Non-local Sparse Image Models

Demosaicking results, Kodak database

More importantly than a PSNR improvement:

Regular sparsity on the left, Joint-sparsity on the right

Structured Sparsity

[Jenatton et al., 2009]

Case of general overlapping groups.

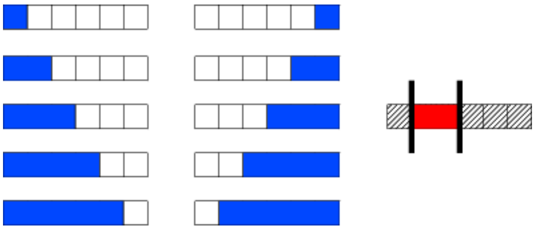
When penalizing by the ℓ_1-ℓ_2 norm,

$$\sum_{G \in \mathcal{G}} \|\boldsymbol{\alpha}_G\|_2 = \sum_{G \in \mathcal{G}} \Big(\sum_{j \in G} \boldsymbol{\alpha}_j^2 \Big)^{1/2}$$

- The ℓ_1 norm induces sparsity at the group level:
  - Some $\boldsymbol{\alpha}_G$'s are set to zero.
- Inside the groups, the ℓ_2 norm does not promote sparsity.
- Intuitively, variables belonging to the same groups are encouraged to be set to zero together.
- Optimization via reweighted least-squares, proximal methods, etc...

Examples of set of groups $\mathcal{G}$ (1/3)

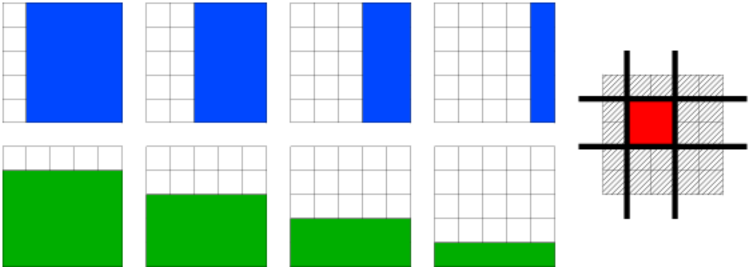
Selection of contiguous patterns on a sequence, $p = 6$.



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

Examples of set of groups $\mathcal{G}$ (2/3)

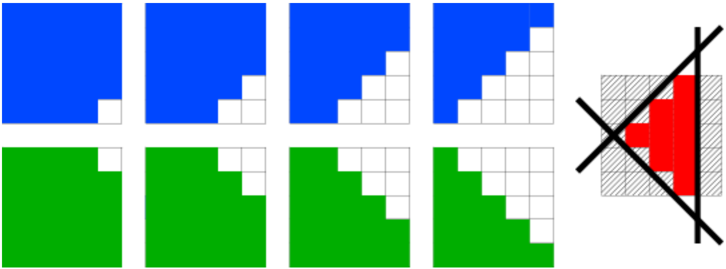
Selection of rectangles on a 2-D grids, $p = 25$.



- $\mathcal{G}$ is the set of blue/green groups (with their not displayed complements).
- Any union of blue/green groups set to zero leads to the selection of a rectangle.

Examples of set of groups $\mathcal{G}$ (3/3)

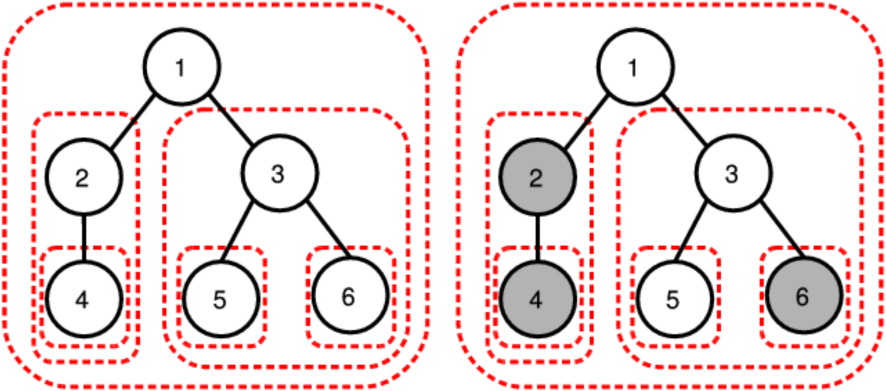
Selection of diamond-shaped patterns on a 2-D grids, $p = 25$.



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

Hierarchical Norms

[Jenatton, Mairal, Obozinski, and Bach, 2010a]



A node can be active only if its **ancestors are active**.

The selected patterns are **rooted subtrees.**

Optimization via efficient proximal methods (same cost as ℓ_1)

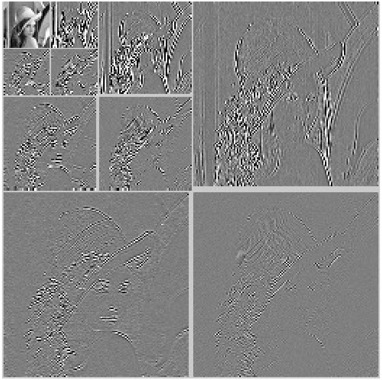
Wavelet denoising with hierarchical norms

[Jenatton, Mairal, Obozinski, and Bach, 2010b]

Classical wavelet denoising [Donoho and Johnstone, 1995]:

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

When **D** is orthogonal, the solution is obtained via **soft-thresholding**.



Wavelet denoising with hierarchical norms

[Jenatton, Mairal, Obozinski, and Bach, 2010b]

Wavelet with hierarchical norm: Add **a-priori knowledge** that the coefficients are embedded in a tree.

(g) Barb., $\sigma = 50$, ℓ_1

(h) Barb., $\sigma = 50$, tree

Wavelet denoising with hierarchical norms

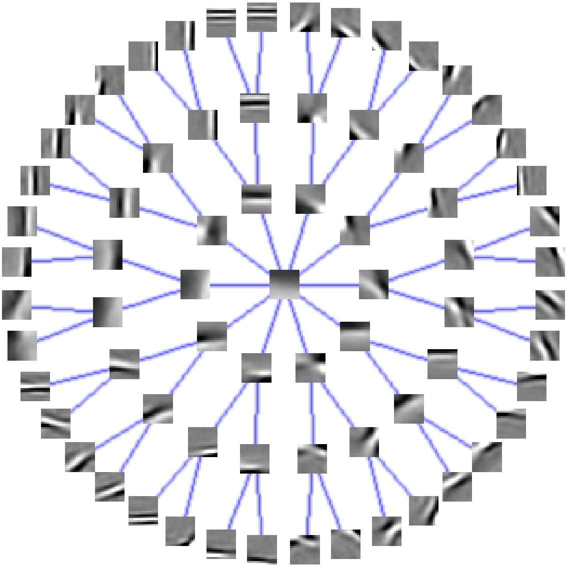
[Jenatton, Mairal, Obozinski, and Bach, 2010b]

Benchmark on a database of 12 **standard images:**

		Haar			
	σ	ℓ_0	ℓ_1	Ω_{ℓ_2}	Ω_{ℓ_∞}
PSNR	5	34.48	35.52	**35.89**	35.79
	10	29.63	30.74	**31.40**	31.23
	25	24.44	25.30	**26.41**	26.14
	50	21.53	20.42	**23.41**	23.05
	100	19.27	19.43	**20.97**	20.58
IPSNR	5	-	$1.04 \pm .31$	$\mathbf{1.41} \pm \mathbf{.45}$	$1.31 \pm .41$
	10	-	$1.10 \pm .22$	$\mathbf{1.76} \pm \mathbf{.26}$	$1.59 \pm .22$
	25	-	$.86 \pm .35$	$\mathbf{1.96} \pm \mathbf{.22}$	$1.69 \pm .21$
	50	-	$.46 \pm .28$	$\mathbf{1.87} \pm \mathbf{.20}$	$1.51 \pm .20$
	100	-	$.15 \pm .23$	$\mathbf{1.69} \pm \mathbf{.19}$	$1.30 \pm .19$

Hierarchical Dictionaries

[Jenatton, Mairal, Obozinski, and Bach, 2010a]

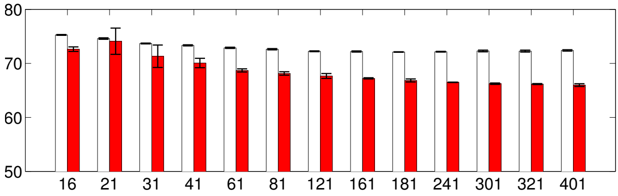


Application to patch reconstrution

[Jenatton, Mairal, Obozinski, and Bach, 2010a]

- Reconstruction of 100,000 8×8 natural images patches
  - Remove randomly subsampled pixels
  - Reconstruct with matrix factorization and structured sparsity

noise	50 %	60 %	70 %	80 %	90 %
flat	19.3 ± 0.1	26.8 ± 0.1	36.7 ± 0.1	50.6 ± 0.0	72.1 ± 0.0
tree	18.6 ± 0.1	25.7 ± 0.1	35.0 ± 0.1	48.0 ± 0.0	65.9 ± 0.3

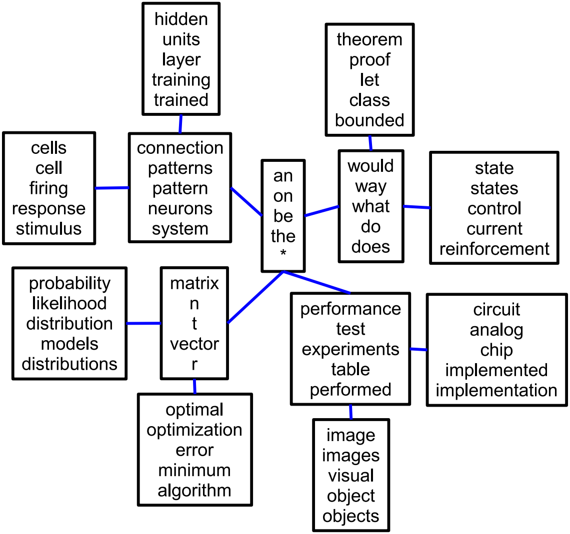


Hierarchical Topic Models for text corpora

[Jenatton, Mairal, Obozinski, and Bach, 2010a]

- Each document is modeled through word counts
- Low-rank matrix factorization of word-document matrix
- Probabilistic topic models such as Latent Dirichlet Allocation [Blei et al., 2003]
- Organise the topics in a tree.
- Previously approached using non-parametric Bayesian methods (Hierarchical Chinese Restaurant Process and nested Dirichlet Process): [Blei et al., 2010]
- **Can we achieve similar performance with simple matrix factorization formulation?**

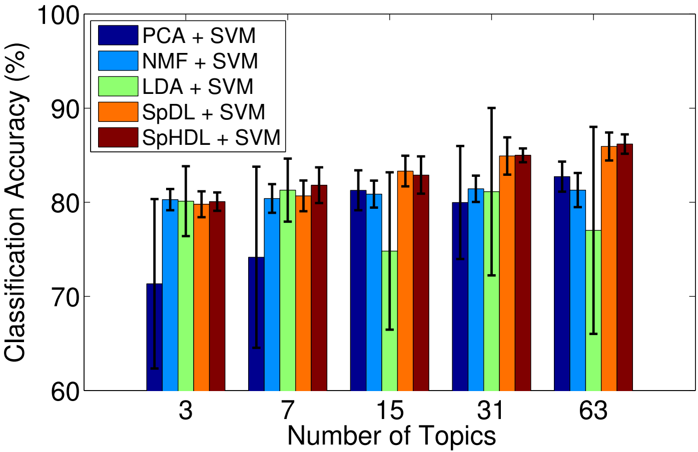
Tree of Topics



Classification based on topics

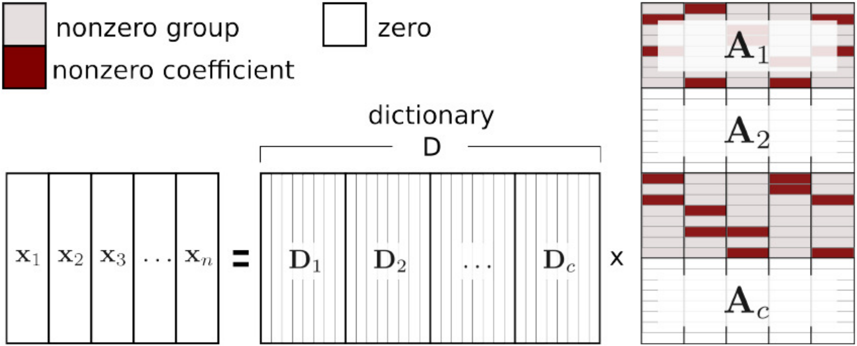
Comparison on predicting newsgroup article subjects

- 20 newsgroup articles (1425 documents, 13312 words)



Group Lasso + Sparsity

[Sprechmann et al., 2010a]



Important messages

- The ℓ_1-norm induces sparsity and shrinks the coefficients (soft-thresholding)
- The regularization path of the Lasso is piecewise linear.
- Learning the dictionary is simple, fast and scalable.
- Dictionary learning is related to several matrix factorization problems.
- Sparsity can be induced at the group level.
- Structured sparsity opens a whole range of new applications.

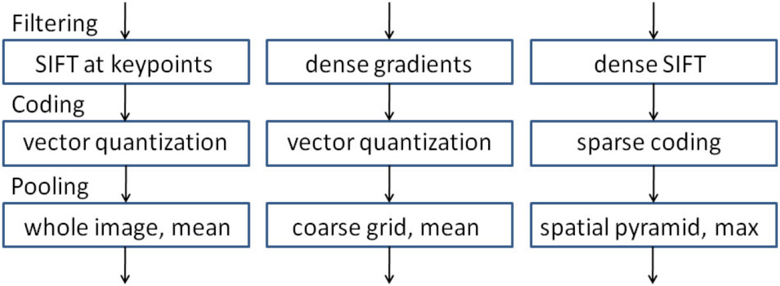
Software SPAMS is available for all of this:
`www.di.ens.fr/willow/SPAMS/`.

Next topics: Computer Vision

- Intriguing results on the use of dictionary learning for bags of words.
- Modelling the local appearance of image patches.
- Preliminary applications of structured sparsity.

1. Image Processing Applications
2. Sparse Linear Models and Dictionary Learning
3. Computer Vision Applications
   - Learning codebooks for image classification
   - Modelling the local appearance of image patches
   - Background subtraction with structured sparsity
4. Optimization for sparse methods

Learning Codebooks for Image Classification



Idea

Replacing Vector Quantization by Learned Dictionaries!

- unsupervised: [Yang et al., 2009]
- supervised: [Boureau et al., 2010, Yang et al., 2010]

Learning Codebooks for Image Classification

Let an image be represented by a set of low-level descriptors $\mathbf{y}_i$ at N locations identified with their indices $i = 1, \ldots, N$.

- hard-quantization:

$$\mathbf{y}_i \approx \mathbf{D}\boldsymbol{\alpha}_i, \quad \boldsymbol{\alpha}_i \in \{0,1\}^p \text{ and } \sum_{j=1}^{p} \boldsymbol{\alpha}_i[j] = 1$$

- soft-quantization:

$$\boldsymbol{\alpha}_i[j] = \frac{e^{-\beta\|\mathbf{y}_i - \mathbf{d}_j\|_2^2}}{\sum_{k=1}^{p} e^{-\beta\|\mathbf{y}_i - \mathbf{d}_k\|_2^2}}$$

- sparse coding:

$$\mathbf{y}_i \approx \mathbf{D}\boldsymbol{\alpha}_i, \quad \boldsymbol{\alpha}_i = \arg\min_{\boldsymbol{\alpha}} \frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$$

Learning Codebooks for Image Classification

Table from Boureau et al. [2010]

Method	Caltech-101, 30 training examples		15 Scenes, 100 training examples	
	Average Pool	Max Pool	Average Pool	Max Pool
	Results with basic features, SIFT extracted each 8 pixels			
Hard quantization, linear kernel	51.4 ± 0.9 [256]	64.3 ± 0.9 [256]	73.9 ± 0.9 [1024]	80.1 ± 0.6 [1024]
Hard quantization, intersection kernel	64.2 ± 1.0 [256] **(1)**	64.3 ± 0.9 [256]	80.8 ± 0.4 [256] **(1)**	80.1 ± 0.6 [1024]
Soft quantization, linear kernel	57.9 ± 1.5 [1024]	69.0 ± 0.8 [256]	75.6 ± 0.5 [1024]	81.4 ± 0.6 [1024]
Soft quantization, intersection kernel	66.1 ± 1.2 [512] **(2)**	70.6 ± 1.0 [1024]	81.2 ± 0.4 [1024] **(2)**	83.0 ± 0.7 [1024]
Sparse codes, linear kernel	61.3 ± 1.3 [1024]	$\mathbf{71.5 \pm 1.1}$ [1024] **(3)**	76.9 ± 0.6 [1024]	83.1 ± 0.6 [1024] **(3)**
Sparse codes, intersection kernel	70.3 ± 1.3 [1024]	$\mathbf{71.8 \pm 1.0}$ [1024] **(4)**	83.2 ± 0.4 [1024]	$\mathbf{84.1 \pm 0.5}$ [1024] **(4)**
	Results with macrofeatures and denser SIFT sampling			
Hard quantization, linear kernel	55.6 ± 1.6 [256]	70.9 ± 1.0 [1024]	74.0 ± 0.5 [1024]	80.1 ± 0.5 [1024]
Hard quantization, intersection kernel	68.8 ± 1.4 [512]	70.9 ± 1.0 [1024]	81.0 ± 0.5 [1024]	80.1 ± 0.5 [1024]
Soft quantization, linear kernel	61.6 ± 1.6 [1024]	71.5 ± 1.0 [1024]	76.4 ± 0.7 [1024]	81.5 ± 0.4 [1024]
Soft quantization, intersection kernel	70.1 ± 1.3 [1024]	73.2 ± 1.0 [1024]	81.8 ± 0.4 [1024]	83.0 ± 0.4 [1024]
Sparse codes, linear kernel	65.7 ± 1.4 [1024]	$\mathbf{75.1 \pm 0.9}$ [1024]	78.2 ± 0.7 [1024]	83.6 ± 0.4 [1024]
Sparse codes, intersection kernel	73.7 ± 1.3 [1024]	$\mathbf{75.7 \pm 1.1}$ [1024]	83.5 ± 0.4 [1024]	$\mathbf{84.3 \pm 0.5}$ [1024]

	Unsup	Discr
Linear	83.6 ± 0.4	$\mathbf{84.9 \pm 0.3}$
Intersect	84.3 ± 0.5	$\mathbf{84.7 \pm 0.4}$

Yang et al. [2009] have won the PASCAL VOC'09 challenge using this kind of techniques.

Learning dictionaries with a discriminative cost function

Idea:

Let us consider 2 sets S_-, S_+ of signals representing 2 different classes. Each set should admit a dictionary best adapted to its reconstruction.

Classification procedure for a signal $\mathbf{y} \in \mathbb{R}^n$:

$$\min(\mathbf{R}^\star(\mathbf{y}, \mathbf{D}_-), \mathbf{R}^\star(\mathbf{y}, \mathbf{D}_+))$$

where

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

"Reconstructive" training

$$\begin{cases} \min_{\mathbf{D}_-} \sum_{i \in S_-} \mathbf{R}^\star(\mathbf{y}_i, \mathbf{D}_-) \\ \min_{\mathbf{D}_+} \sum_{i \in S_+} \mathbf{R}^\star(\mathbf{y}_i, \mathbf{D}_+) \end{cases}$$

[Grosse et al., 2007], [Huang and Aviyente, 2006],
[Sprechmann et al., 2010b] for unsupervised clustering (CVPR '10)

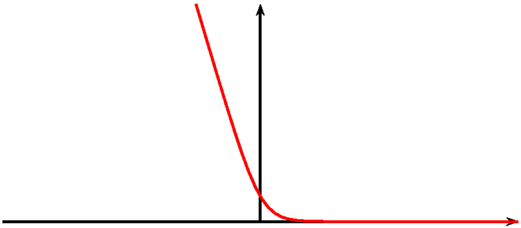
Learning dictionaries with a discriminative cost function

"Discriminative" training

[Mairal, Bach, Ponce, Sapiro, and Zisserman, 2008a]

$$\min_{\mathbf{D}_-,\mathbf{D}_+} \sum_i \mathcal{C}\Big(\lambda z_i \big(\mathbf{R}^\star(\mathbf{y}_i, \mathbf{D}_-) - \mathbf{R}^\star(\mathbf{y}_i, \mathbf{D}_+)\big)\Big),$$

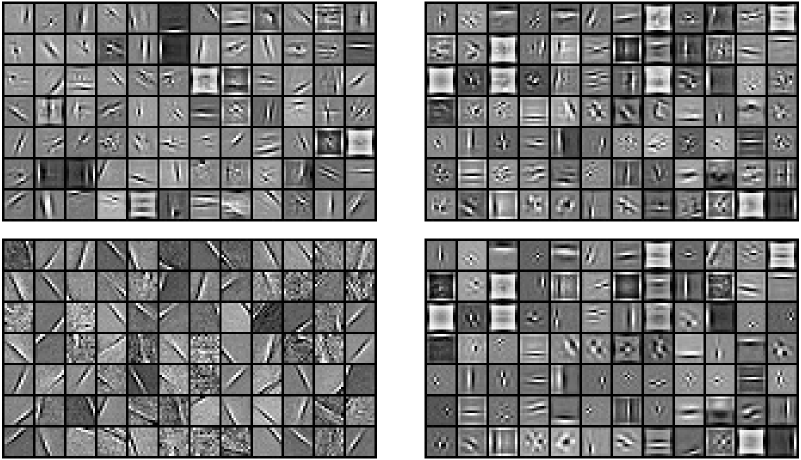
where $z_i \in \{-1, +1\}$ is the label of $\mathbf{y}_i$.



Logistic regression function

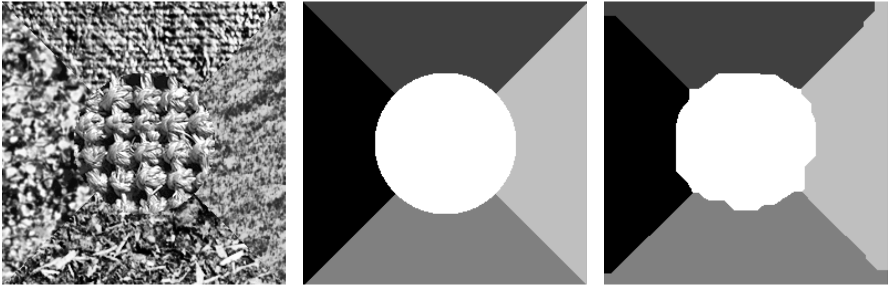
Learning dictionaries with a discriminative cost function

Examples of dictionaries



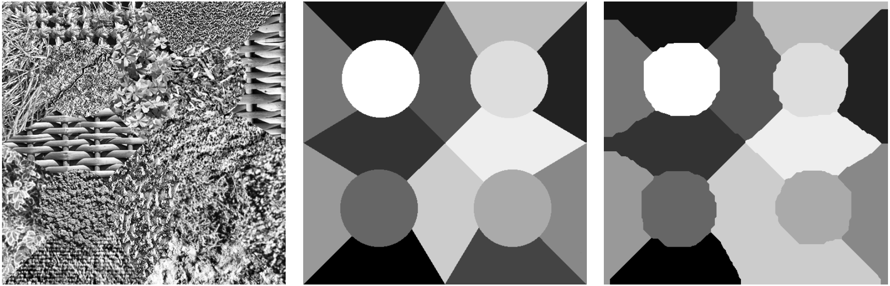
Top: reconstructive, Bottom: discriminative, Left: Bicycle, Right: Background.

Learning dictionaries with a discriminative cost function
Texture segmentation



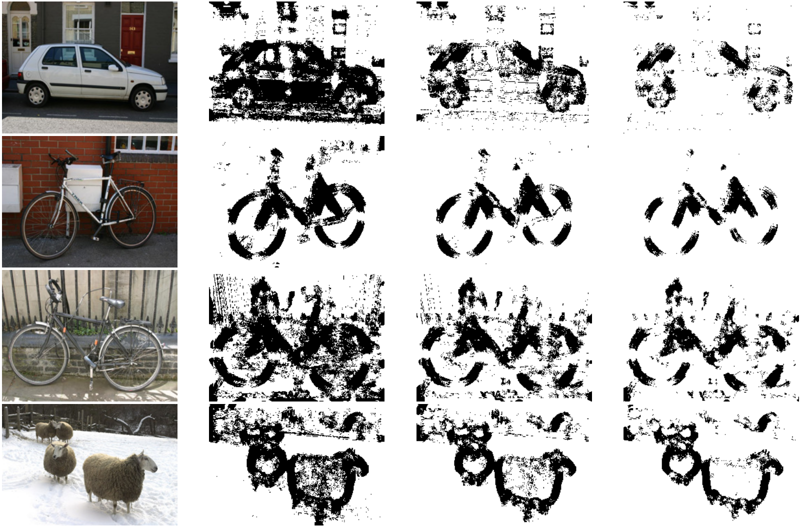
Learning dictionaries with a discriminative cost function

Texture segmentation



Learning dictionaries with a discriminative cost function

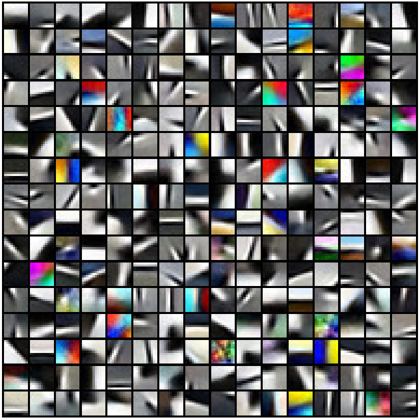
Pixelwise classification



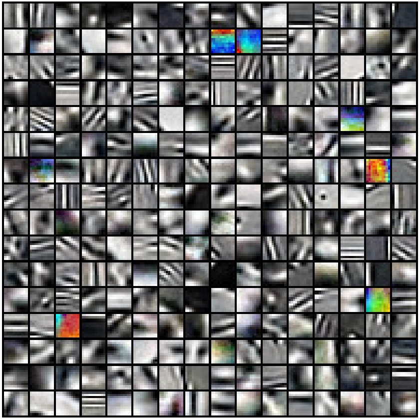
Learning dictionaries with a discriminative cost function
weakly-supervised pixel classification

Application to edge detection and classification

[Mairal, Leordeanu, Bach, Hebert, and Ponce, 2008c]

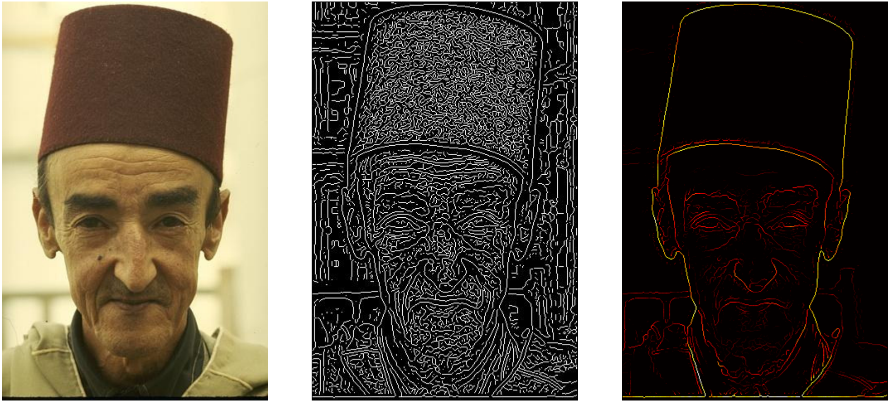


Good edges



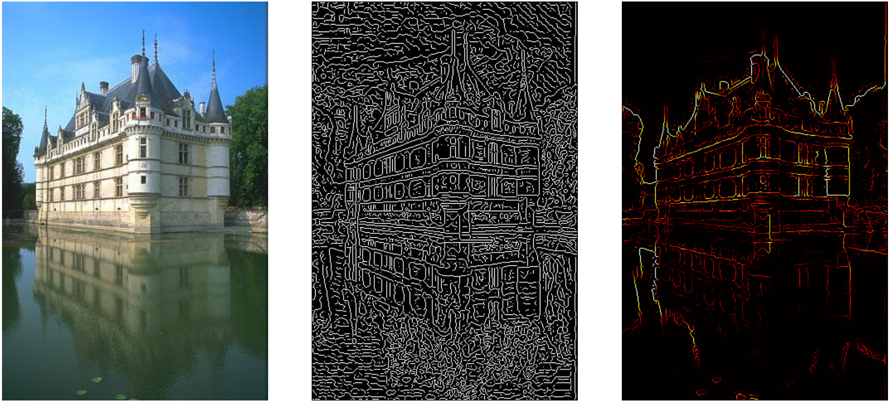
Bad edges

Application to edge detection and classification
Berkeley segmentation benchmark



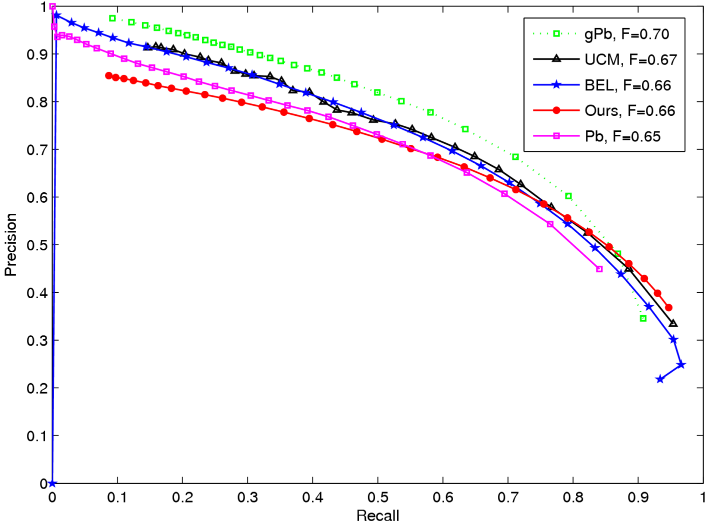
Raw edge detection on the right

Application to edge detection and classification
Berkeley segmentation benchmark



Raw edge detection on the right

Application to edge detection and classification
Berkeley segmentation benchmark

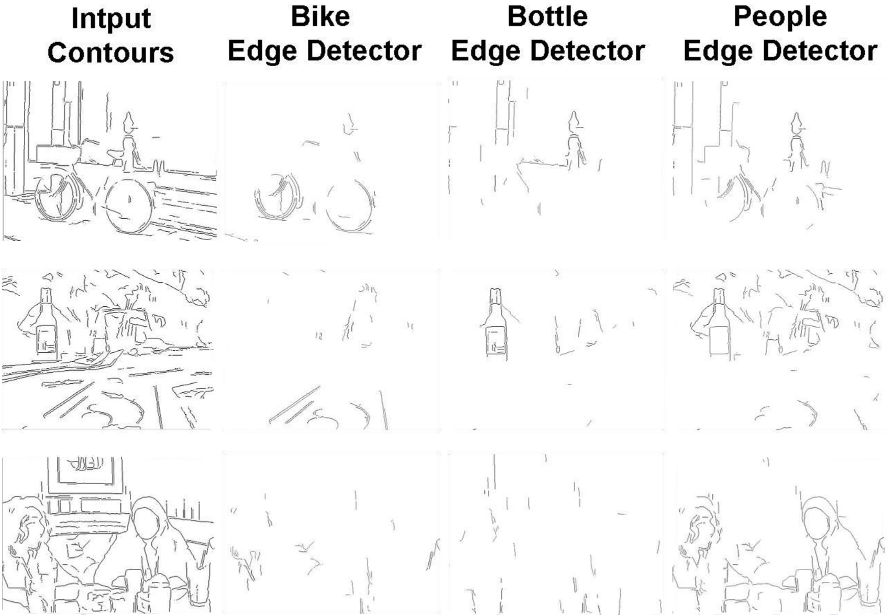


Application to edge detection and classification

Contour-based classifier: [Leordeanu, Hebert, and Sukthankar, 2007]

Is there a bike, a motorbike, a car or a person on this image?

Application to edge detection and classification



Application to edge detection and classification
Performance gain due to the prefiltering

Ours + [Leordeanu '07]	[Leordeanu '07]	[Winn '05]
96.8%	89.4%	76.9%

Recognition rates for the same experiment as [Winn et al., 2005] on VOC 2005.

Category	Ours+[Leordeanu '07]	[Leordeanu '07]
Aeroplane	71.9%	61.9%
Boat	67.1%	56.4%
Cat	82.6%	53.4%
Cow	68.7%	59.2%
Horse	76.0%	67%
Motorbike	80.6%	73.6%
Sheep	72.9%	58.4%
Tvmonitor	87.7%	83.8%
Average	75.9%	64.2 %

Recognition performance at equal error rate for 8 classes on a subset of images from Pascal 07.

Digital Art Authentification

Data Courtesy of Hugues, Graham, and Rockmore [2009]

Authentic

Fake

Given a pair of paintings, Which one is the fake?

Digital Art Authentification

Data Courtesy of Hugues, Graham, and Rockmore [2009]

Authentic

Fake

Fake

Digital Art Authentification

Data Courtesy of Hugues, Graham, and Rockmore [2009]

Authentic

Fake

Authentic

Background Subtraction

Given a video sequence, how can we remove foreground objects?

video sequence 1

video sequence 2

Background Subtraction

$$\underbrace{\mathbf{y}}_{\text{frame}} \approx \underbrace{\mathbf{D}\boldsymbol{\alpha}}_{\text{linear combination of background frames}} + \underbrace{\mathbf{e}}_{\text{error term}}.$$

Solved by

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^p, \mathbf{e}\in\mathbb{R}^m} \frac{1}{2}\|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha} - \mathbf{e}\|_2^2 + \lambda_1\|\boldsymbol{\alpha}\| + \lambda_2\psi(\mathbf{e}).$$

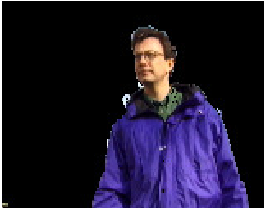
Same idea used by Wright et al. '09 for robust face recognition with $\psi = \ell_1$.

Background Subtraction

(a) input

(b) estimated background

(c) foreground, ℓ_1

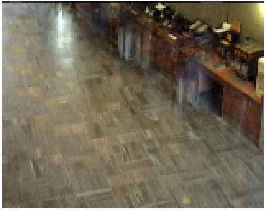


(d) foreground, ℓ_1+struct

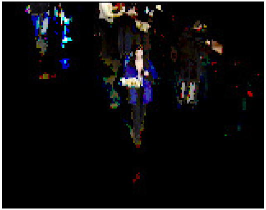
(e) other example

Background Subtraction

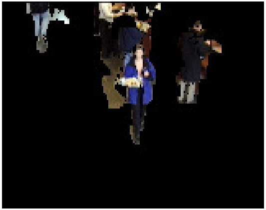
(a) input



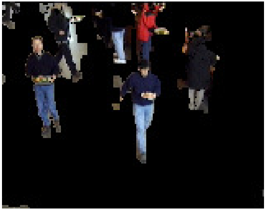
(b) estimated background



(c) foreground, ℓ_1



(d) foreground, ℓ_1+struct



(e) other example

Important messages

- Learned dictionaries are well adapted to model the local appearance of images and edges.
- They can be used to learn dictionaries of SIFT features.
- New applications coming with structured sparsity?

Next topics

- Optimization for solving sparse decomposition problems
- Optimization for dictionary learning

Recall: The Sparse Decomposition Problem

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \underbrace{\frac{1}{2}\|\mathbf{y}-\mathbf{D}\boldsymbol{\alpha}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda\psi(\boldsymbol{\alpha})}_{\textbf{sparsity-inducing regularization}}$$

ψ induces sparsity in $\boldsymbol{\alpha}$. It can be

- the ℓ_0 "pseudo-norm". $\|\boldsymbol{\alpha}\|_0 \triangleq \#\{i \text{ s.t. } \boldsymbol{\alpha}[i] \neq 0\}$ (NP-hard)
- the ℓ_1 norm. $\|\boldsymbol{\alpha}\|_1 \triangleq \sum_{i=1}^{p} |\boldsymbol{\alpha}[i]|$ (convex)
- ...

This is a selection problem.

Finding your way in the sparse coding literature...

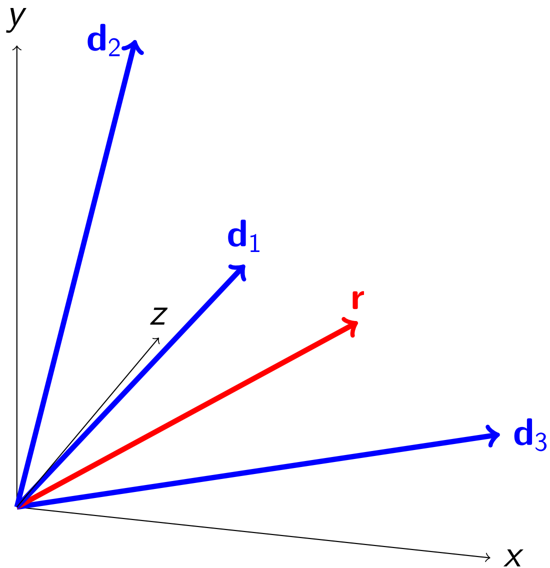
...is not easy. The literature is vast, redundant, sometimes confusing and many papers are claiming victory...

The main class of methods are

- greedy procedures [Mallat and Zhang, 1993], [Weisberg, 1980]
- homotopy [Osborne et al., 2000], [Efron et al., 2004], [Markowitz, 1956]
- soft-thresholding based methods [Fu, 1998], [Daubechies et al., 2004], [Friedman et al., 2007], [Nesterov, 2007], [Beck and Teboulle, 2009], ...
- reweighted-ℓ_2 methods [Daubechies et al., 2009],...
- active-set methods [Roth and Fischer, 2008].
- ...

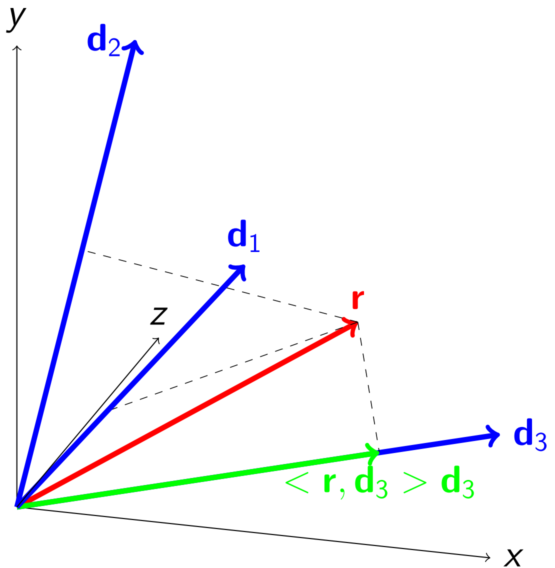
Matching Pursuit

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



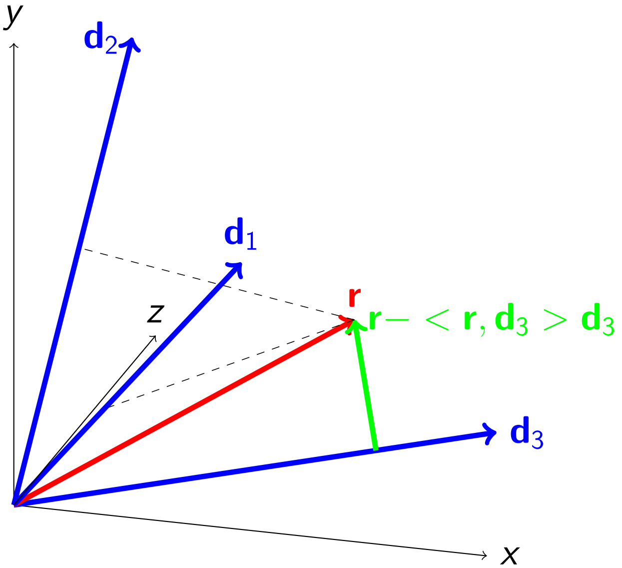
Matching Pursuit

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



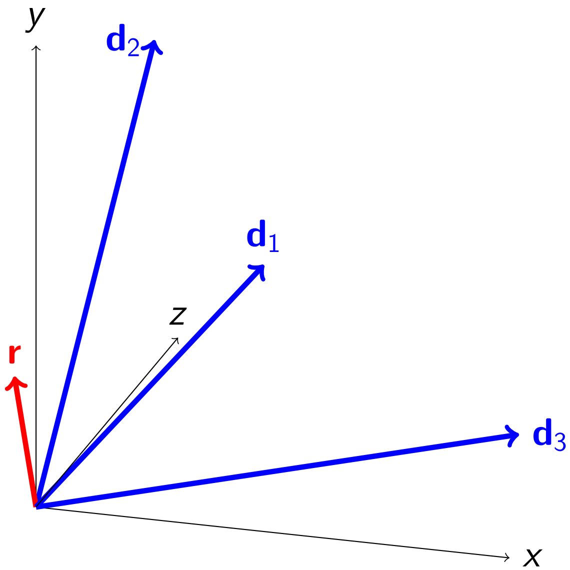
Matching Pursuit

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



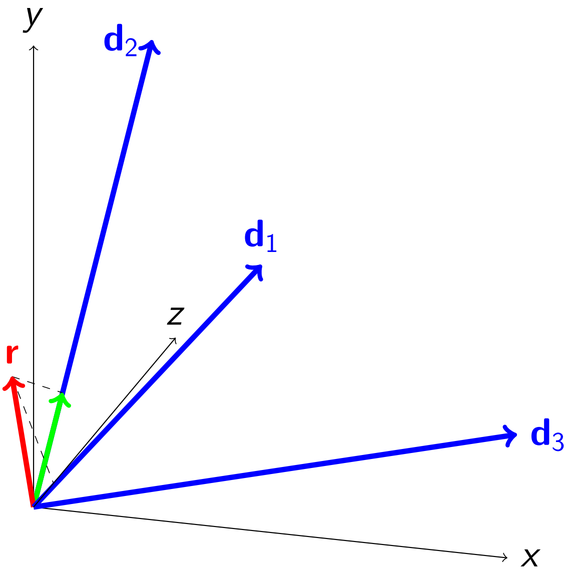
Matching Pursuit

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



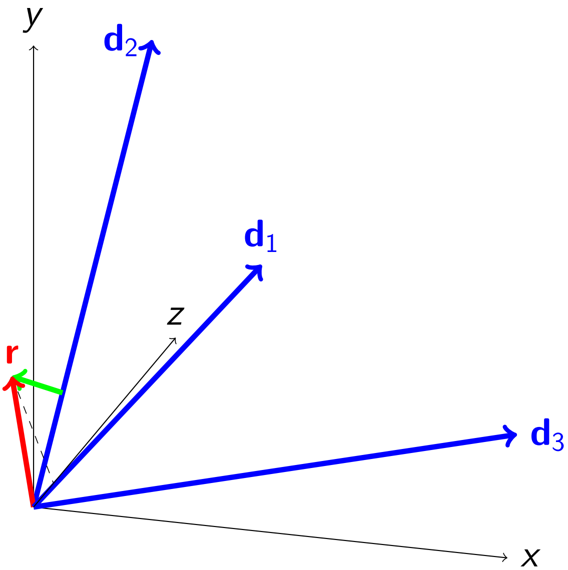
Matching Pursuit

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



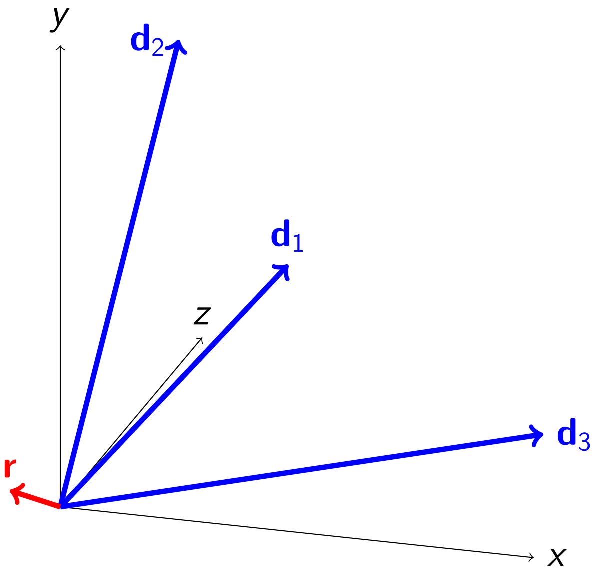
Matching Pursuit

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



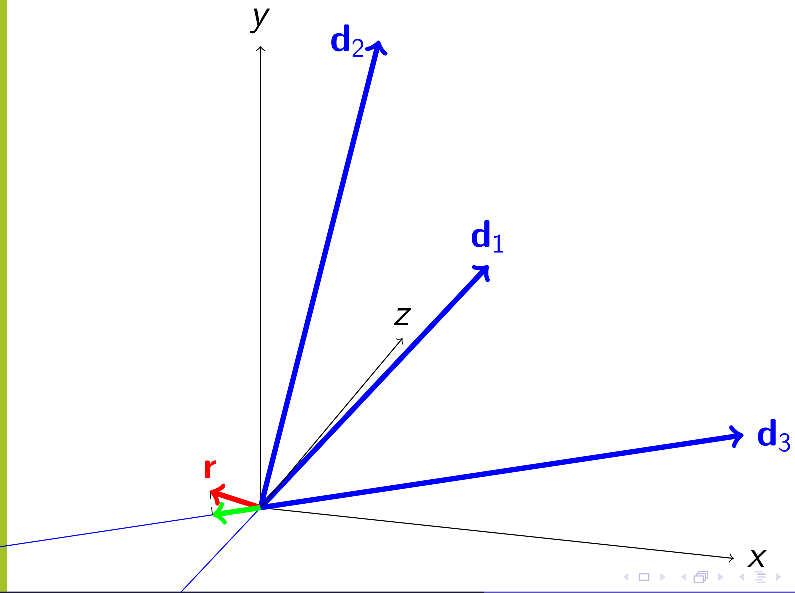
Matching Pursuit

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



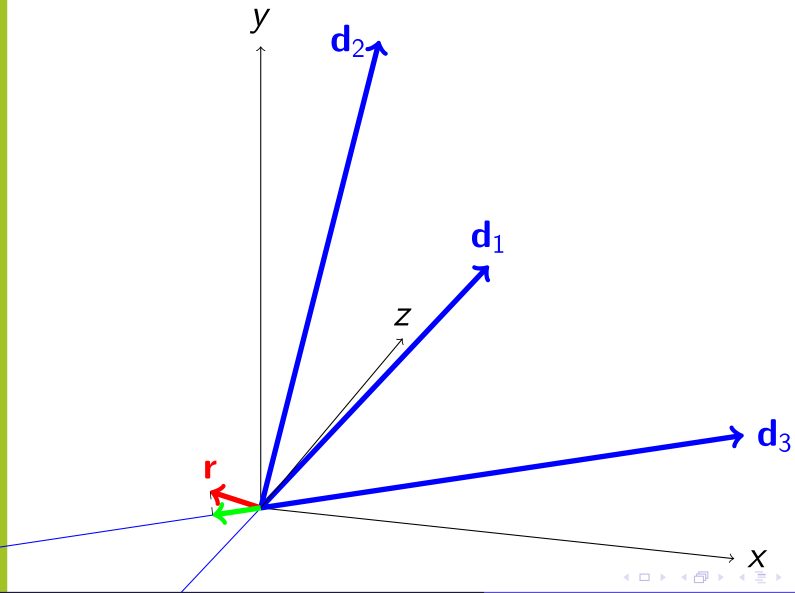
Matching Pursuit

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



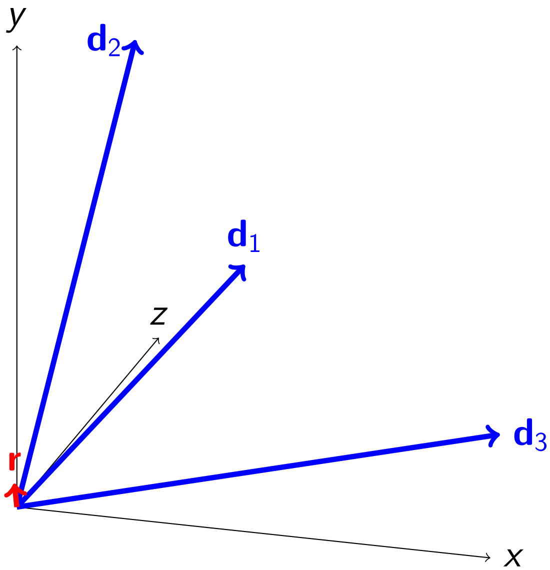
Matching Pursuit

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



Matching Pursuit

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



Matching Pursuit

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

1: $\boldsymbol{\alpha} \leftarrow 0$

2: $\mathbf{r} \leftarrow \mathbf{y}$ (residual).

3: **while** $\|\boldsymbol{\alpha}\|_0 < L$ **do**

4: Select the atom with maximum correlation with the residual

$$\hat{\imath} \leftarrow \arg\max_{i=1,\ldots,p} |\mathbf{d}_i^T \mathbf{r}|$$

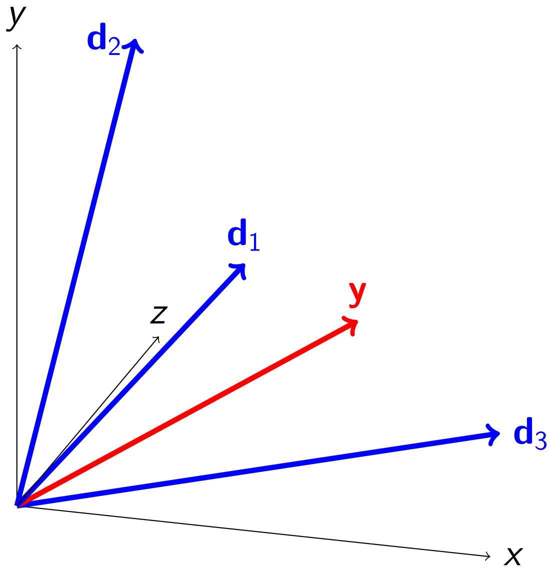
5: Update the residual and the coefficients

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

6: **end while**

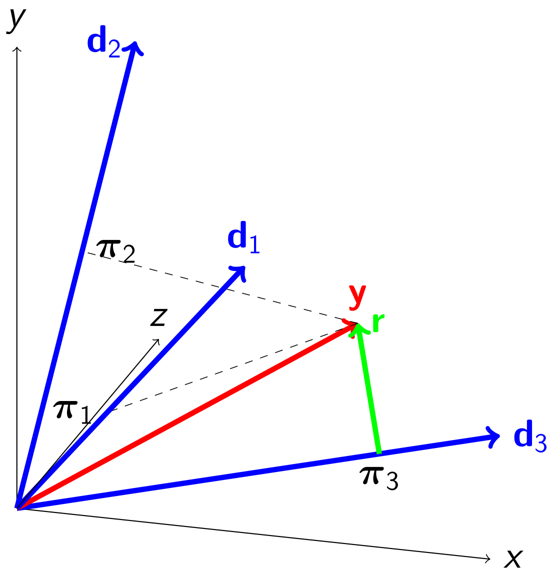
Orthogonal Matching Pursuit

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



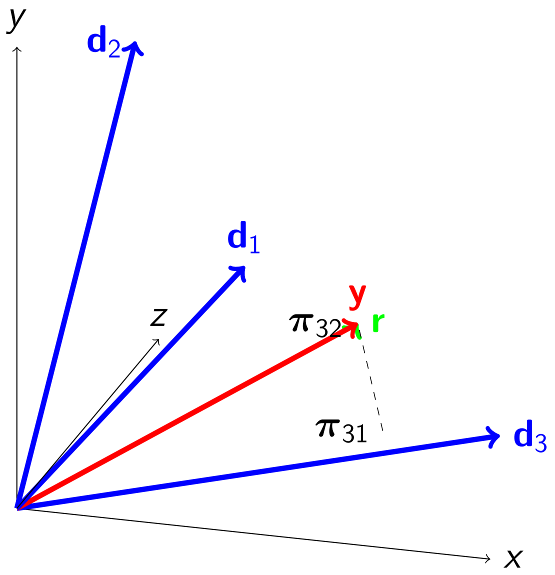
Orthogonal Matching Pursuit

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



Orthogonal Matching Pursuit

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



Orthogonal Matching Pursuit

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

$\Gamma = \emptyset$.
for $iter = 1, \ldots, L$ **do**
    Select the atom which most reduces the objective

$$\hat{\imath} \leftarrow \arg\min_{i \in \Gamma^C} \left\{ \min_{\boldsymbol{\alpha}'} \|\mathbf{y} - \mathbf{D}_{\Gamma \cup \{i\}} \boldsymbol{\alpha}'\|_2^2 \right\}$$

    Update the active set: $\Gamma \leftarrow \Gamma \cup \{\hat{\imath}\}$.
    Update the residual (orthogonal projection)

$$\mathbf{r} \leftarrow (\mathbf{I} - \mathbf{D}_\Gamma (\mathbf{D}_\Gamma^T \mathbf{D}_\Gamma)^{-1} \mathbf{D}_\Gamma^T)\mathbf{y}.$$

    Update the coefficients

$$\boldsymbol{\alpha}_\Gamma \leftarrow (\mathbf{D}_\Gamma^T \mathbf{D}_\Gamma)^{-1} \mathbf{D}_\Gamma^T \mathbf{y}.$$

end for

Orthogonal Matching Pursuit

Contrary to MP, an atom can only be selected one time with OMP. It is, however, more difficult to implement efficiently. The keys for a good implementation in the case of a large number of signals are

- Precompute the Gram matrix $\mathbf{G} = \mathbf{D}^T\mathbf{D}$ once in for all,
- Maintain the computation of $\mathbf{D}^T\mathbf{r}$ for each signal,
- Maintain a Cholesky decomposition of $(\mathbf{D}_\Gamma^T\mathbf{D}_\Gamma)^{-1}$ for each signal.

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

$$\underbrace{O(p^2m)}_{\text{Gram matrix}} + \underbrace{O(nL^3)}_{\text{Cholesky}} + \underbrace{O(n(pm+pL^2))}_{\mathbf{D}^T\mathbf{r}} = O(np(m+L^2))$$

It is also possible to use the matrix inversion lemma instead of a Cholesky decomposition (same complexity, but less numerical stability)

Example with the software SPAMS

Software available at http://www.di.ens.fr/willow/SPAMS/

```
>> 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 a 8-cores 2.83Ghz machine: **230000 signals processed per second!**

Optimality conditions of the Lasso

Nonsmooth optimization

Directional derivatives and subgradients are useful tools for studying ℓ_1-decomposition problems:

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

In this tutorial, we use the **directional derivatives** to derive simple optimality conditions of the Lasso.

For more information on convex analysis and nonsmooth optimization, see the following books: [Boyd and Vandenberghe, 2004], [Nocedal and Wright, 2006], [Borwein and Lewis, 2006], [Bonnans et al., 2006], [Bertsekas, 1999].

Optimality conditions of the Lasso

Directional derivatives

- **Directional derivative** in the direction $\mathbf{u}$ at $\boldsymbol{\alpha}$:

$$\nabla f(\boldsymbol{\alpha}, \mathbf{u}) = \lim_{t \to 0^+} \frac{f(\boldsymbol{\alpha} + t\mathbf{u}) - f(\boldsymbol{\alpha})}{t}$$

- Main idea: in non smooth situations, one may need to look at all directions $\mathbf{u}$ and not simply p independent ones!
- **Proposition 1:** if f is differentiable in $\boldsymbol{\alpha}$, $\nabla f(\boldsymbol{\alpha}, \mathbf{u}) = \nabla f(\boldsymbol{\alpha})^T \mathbf{u}$.
- **Proposition 2:** $\boldsymbol{\alpha}$ is optimal iff for all $\mathbf{u}$ in $\mathbb{R}^p$, $\nabla f(\boldsymbol{\alpha}, \mathbf{u}) \geq 0$.

Optimality conditions of the Lasso

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

$\boldsymbol{\alpha}^\star$ is optimal iff for all $\mathbf{u}$ in $\mathbb{R}^p$, $\nabla f(\boldsymbol{\alpha}, \mathbf{u}) \geq 0$—that is,

$$-\mathbf{u}^T\mathbf{D}^T(\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^\star) + \lambda \sum_{i, \boldsymbol{\alpha}^\star[i] \neq 0} \text{sign}(\boldsymbol{\alpha}^\star[i])\mathbf{u}[i] + \lambda \sum_{i, \boldsymbol{\alpha}^\star[i] = 0} |\mathbf{u}_i| \geq 0,$$

which is equivalent to the following conditions:

$$\forall i = 1, \ldots, p, \quad \begin{cases} |\mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^\star)| & \leq & \lambda & \text{if } \boldsymbol{\alpha}^\star[i] = 0 \\ \mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^\star) & = & \lambda \, \text{sign}(\boldsymbol{\alpha}^\star[i]) & \text{if } \boldsymbol{\alpha}^\star[i] \neq 0 \end{cases}$$

Homotopy

- A homotopy method provides a set of solutions indexed by a parameter.
- The regularization path $(\lambda, \boldsymbol{\alpha}^\star(\lambda))$ for instance!!
- It can be useful when the path has some “nice” properties (piecewise linear, piecewise quadratic).
- LARS [Efron et al., 2004] starts from a trivial solution, and follows the regularization path of the Lasso, which is is **piecewise linear**.

Homotopy, LARS

[Osborne et al., 2000], [Efron et al., 2004]

$$\forall i = 1, \ldots, p, \quad \begin{cases} |\mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^\star)| & \leq & \lambda & \text{if } \boldsymbol{\alpha}^\star[i] = 0 \\ \mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^\star) & = & \lambda \, \text{sign}(\boldsymbol{\alpha}^\star[i]) & \text{if } \boldsymbol{\alpha}^\star[i] \neq 0 \end{cases} \tag{5}$$

The regularization path is piecewise linear:

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

A simple interpretation of LARS

- Start from the trivial solution ($\lambda = \|\mathbf{D}^T \mathbf{y}\|_\infty, \boldsymbol{\alpha}^\star(\lambda) = 0$).
- Maintain the computations of $|\mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}^\star(\lambda))|$ for all i.
- Maintain the computation of the current direction $\mathbf{B}$.
- Follow the path by reducing λ until the next kink.

Example with the software SPAMS

http://www.di.ens.fr/willow/SPAMS/

```
>> I=double(imread('data/lena.eps'))/255;
>> %extract all patches of I
>> X=normalize(im2col(I,[8 8],'sliding'));
>> %load a dictionary of size 64 x 256
>> D=load('dict.mat');
>>
>> %set the sparsity parameter lambda to 0.15
>> param.lambda=0.15;
>> alpha=mexLasso(X,D,param);
```

On a 8-cores 2.83Ghz machine: **77000 signals processed per second!**
Note that it can also solve **constrained** version of the problem. The complexity is more or less the same as OMP and uses the same tricks (Cholesky decomposition).

Coordinate Descent

- Coordinate descent + nonsmooth objective: **WARNING: not convergent in general**
- Here, the problem is equivalent to a convex smooth optimization problem with **separable** constraints

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

- For this **specific** problem, coordinate descent is **convergent**.
- Supposing $\|\mathbf{d}_i\|_2 = 1$, updating the coordinate i:

$$\boldsymbol{\alpha}[i] \leftarrow \arg\min_{\beta} \frac{1}{2}\|\underbrace{\mathbf{y} - \sum_{j\neq i}\boldsymbol{\alpha}[j]\mathbf{d}_j}_{\mathbf{r}} - \beta\mathbf{d}_i\|_2^2 + \lambda|\beta|$$

$$\leftarrow \text{sign}(\mathbf{d}_i^T\mathbf{r})(|\mathbf{d}_i^T\mathbf{r}| - \lambda)^+$$

- $\Rightarrow$ **soft-thresholding**!

Example with the software SPAMS

http://www.di.ens.fr/willow/SPAMS/

```
>> I=double(imread('data/lena.eps'))/255;
>> %extract all patches of I
>> X=normalize(im2col(I,[8 8],'sliding'));
>> %load a dictionary of size 64 x 256
>> D=load('dict.mat');
>>
>> %set the sparsity parameter lambda to 0.15
>> param.lambda=0.15;
>> param.tol=1e-2;
>> param.itermax=200;
>> alpha=mexCD(X,D,param);
```

On a 8-cores 2.83Ghz machine: **93000 signals processed per second!**

First-order/proximal methods

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

- f is strictly convex and differentiable with a Lipshitz gradient.
- Generalizes the idea of gradient descent

$$\boldsymbol{\alpha}^{k+1} \leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \underbrace{f(\boldsymbol{\alpha}^k) + \nabla f(\boldsymbol{\alpha}^k)^\top (\boldsymbol{\alpha} - \boldsymbol{\alpha}^k)}_{\text{linear approximation}} + \underbrace{\frac{L}{2} \|\boldsymbol{\alpha} - \boldsymbol{\alpha}^k\|_2^2}_{\text{quadratic term}} + \lambda \Omega(\boldsymbol{\alpha})$$

$$\leftarrow \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2} \|\boldsymbol{\alpha} - (\boldsymbol{\alpha}^k - \frac{1}{L} \nabla f(\boldsymbol{\alpha}^k))\|_2^2 + \frac{\lambda}{L} \Omega(\boldsymbol{\alpha})$$

When $\lambda = 0$, $\boldsymbol{\alpha}^{k+1} \leftarrow \boldsymbol{\alpha}^k - \frac{1}{L} \nabla f(\boldsymbol{\alpha}^k)$, this is equivalent to a classical gradient descent step.

First-order/proximal methods

- They require solving efficiently the proximal operator

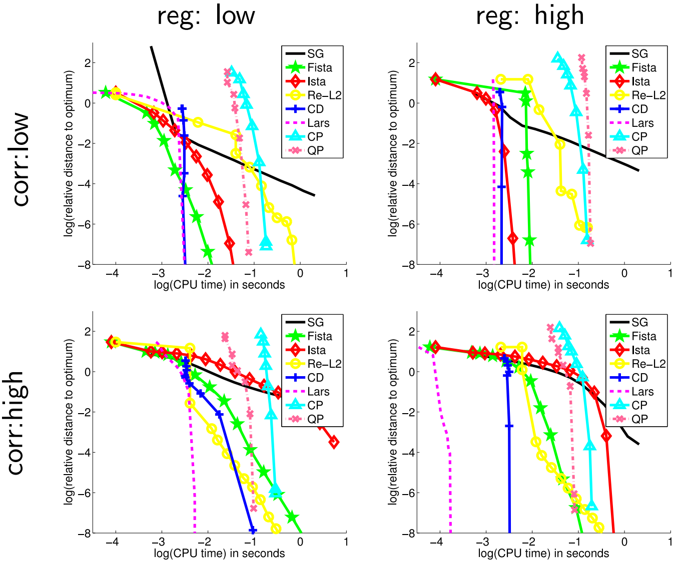
$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{u} - \boldsymbol{\alpha}\|_2^2 + \lambda\Omega(\boldsymbol{\alpha})$$

- For the ℓ_1-norm, this amounts to a soft-thresholding:

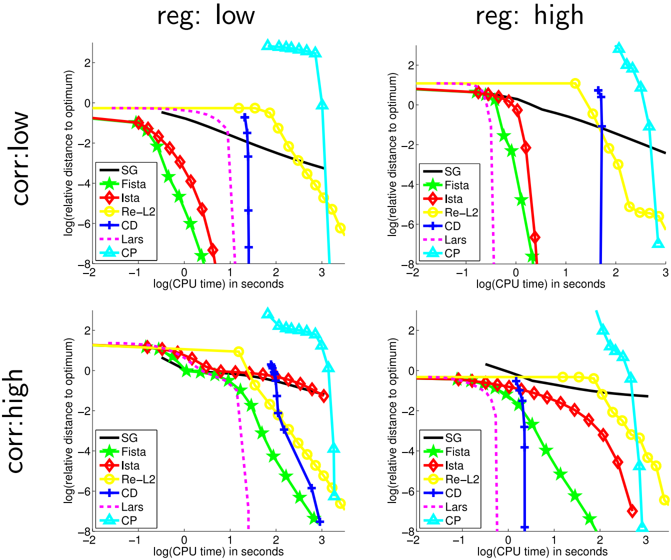
$$\boldsymbol{\alpha}_i^\star = \text{sign}(\mathbf{u}_i)(\mathbf{u}_i - \lambda)^+.$$

- There exists accelerated versions based on Nesterov optimal first-order method (gradient method with "extrapolation") [Beck and Teboulle, 2009, Nesterov, 2007, 1983]
- suited for large-scale experiments.

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



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



Empirical comparison: conclusions

Lasso

- Generic methods very slow
- LARS fastest in **low dimension** or for **high correlation**
- Proximal methods competitive
  - esp. larger setting with weak corr. $+$ weak reg.
- Coordinate descent
  - Dominated by the LARS
  - Would benefit from an offline computation of the matrix

Smooth Losses

- LARS not available $\rightarrow$ CD and proximal methods good candidates

Optimization for Grouped Sparsity

The formulation:

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \underbrace{\frac{1}{2}\|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda \sum_{g \in \mathcal{G}} \|\boldsymbol{\alpha}_g\|_q}_{\textbf{group-sparsity-inducing regularization}}$$

The main class of algorithms for solving grouped-sparsity problems are

- Greedy approaches
- Block-coordinate descent
- Proximal methods

Optimization for Grouped Sparsity

The proximal operator:

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{2}\|\mathbf{u} - \boldsymbol{\alpha}\|_2^2 + \lambda \sum_{g \in \mathcal{G}} \|\boldsymbol{\alpha}_g\|_q$$

For $q = 2$,

$$\boldsymbol{\alpha}_g^\star = \frac{\mathbf{u}_g}{\|\mathbf{u}_g\|_2}(\|\mathbf{u}_g\|_2 - \lambda)^+, \quad \forall g \in \mathcal{G}$$

For $q = \infty$,

$$\boldsymbol{\alpha}_g^\star = \mathbf{u}_g - \Pi_{\|.\|_1 \leq \lambda}[\mathbf{u}_g], \quad \forall g \in \mathcal{G}$$

These formula generalize soft-thrsholding to groups of variables. They are used in block-coordinate descent and proximal algorithms.

Reweighted ℓ_2

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

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

Important messages

- Greedy methods directly address the NP-hard ℓ_0-decomposition problem.
- Homotopy methods can be extremely efficient for small or medium-sized problems, or when the solution is very sparse.
- Coordinate descent provides in general quickly a solution with a small/medium precision, but gets slower when there is a lot of correlation in the dictionary.
- First order methods are very attractive in the large scale setting.
- Other good alternatives exists, active-set, reweighted ℓ_2 methods, stochastic variants, variants of OMP,. . .

Optimization for Dictionary Learning

$$\min_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^{p \times n} \\ \mathbf{D} \in \mathcal{C}}} \sum_{i=1}^{n} \frac{1}{2} \|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1$$

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

- Classical optimization alternates between **D** and $\boldsymbol{\alpha}$.
- Good results, but very slow!

Optimization for Dictionary Learning

[Mairal, Bach, Ponce, and Sapiro, 2009a]

Classical formulation of dictionary learning

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

where

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

Which formulation are we interested in?

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

[Bottou and Bousquet, 2008]: Online learning can

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

Optimization for Dictionary Learning

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

1: $\mathbf{A}_0 = 0$, $\mathbf{B}_0 = 0$.

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

3: Draw $\mathbf{y}_t$

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

5: Aggregate sufficient statistics
$\mathbf{A}_t \leftarrow \mathbf{A}_{t-1} + \boldsymbol{\alpha}_t\boldsymbol{\alpha}_t^T$, $\mathbf{B}_t \leftarrow \mathbf{B}_{t-1} + \mathbf{y}_t\boldsymbol{\alpha}_t^T$

6: Dictionary Update (block-coordinate descent)

$$\mathbf{D}_t \leftarrow \arg\min_{\mathbf{D} \in \mathcal{C}} \frac{1}{t} \sum_{i=1}^{t} \left(\frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1 \right). \quad (6)$$

$$= \arg\min_{\mathbf{D} \in \mathcal{C}} \frac{1}{t} \left(\frac{1}{2} \operatorname{Tr}(\mathbf{D}^T\mathbf{D}\mathbf{A}_t) - \operatorname{Tr}(\mathbf{D}^T\mathbf{B}_t) \right). \quad (7)$$

7: **end for**

Optimization for Dictionary Learning

Which guarantees do we have?

Under a few reasonable assumptions,

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

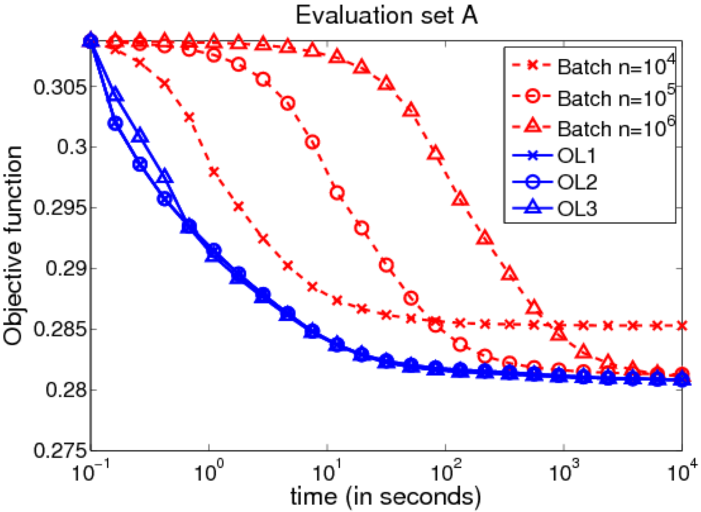
$$\lim_{t \to +\infty} \hat{f}_t(\mathbf{D}_t) - f(\mathbf{D}_t) = 0,$$

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

Extensions (all implemented in SPAMS)

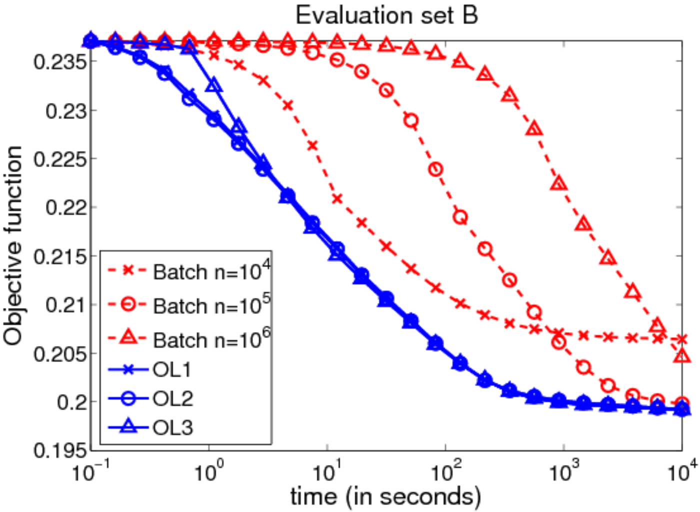
- non-negative matrix decompositions.
- sparse PCA (sparse dictionaries).
- fused-lasso regularizations (piecewise constant dictionaries)

Optimization for Dictionary Learning
Experimental results, batch vs online



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

Optimization for Dictionary Learning
Experimental results, batch vs online



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

References I

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

F. Bach. Bolasso: model consistent lasso estimation through the bootstrap. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

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.

D. P. Bertsekas. *Nonlinear programming*. Athena Scientific Belmont, Mass, 1999.

D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, January 2003.

D. Blei, T. Griffiths, and M. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2): 1–30, 2010.

J.F. Bonnans, J.C. Gilbert, C. Lemarechal, and C.A. Sagastizabal. *Numerical optimization: theoretical and practical aspects*. Springer-Verlag New York Inc, 2006.

References II

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

L. Bottou and O. Bousquet. The trade-offs of large scale learning. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20, pages 161–168. MIT Press, Cambridge, MA, 2008.

Y-L. Boureau, F. Bach, Y. Lecun, and J. Ponce. Learning mid-level features for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

S. P. 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 Multiscale Modelling and Simulation*, 4(2):490–530, 2006.

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.

E. Candes. Compressive sampling. In *Proceedings of the International Congress of Mathematicians*, volume 3, 2006.

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

References III

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.

I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Appl. Math*, 57: 1413–1457, 2004.

I. Daubechies, R. DeVore, M. Fornasier, and S. Gunturk. Iteratively re-weighted least squares minimization for sparse recovery. *Commun. Pure Appl. Math*, 2009.

D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.

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

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

References IV

M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 54(12): 3736–3745, December 2006.

K. Engan, S. O. Aase, and J. H. Husoy. Frame based signal compression using method of optimal directions (MOD). In *Proceedings of the 1999 IEEE International Symposium on Circuits Systems*, volume 4, 1999.

J. Friedman, T. Hastie, H. Hölfling, and R. Tibshirani. Pathwise coordinate optimization. *Annals of statistics*, 1(2):302–332, 2007.

W. J. Fu. Penalized regressions: The bridge versus the Lasso. *Journal of computational and graphical statistics*, 7:397–416, 1998.

R. Grosse, R. Raina, H. Kwong, and A. Y. Ng. Shift-invariant sparse coding for audio classification. In *Proceedings of the Twenty-third Conference on Uncertainty in Artificial Intelligence*, 2007.

BK Gunturk, Y. Altunbasak, and RM Mersereau. Color plane interpolation using alternating projections. *IEEE Transactions on Image Processing*, 11(9):997–1013, 2002.

A. Haar. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 69:331–371, 1910.

References V

K. Huang and S. Aviyente. Sparse representation for signal classification. In *Advances in Neural Information Processing Systems*, Vancouver, Canada, December 2006.

J. M. Hugues, D. J. Graham, and D. N. Rockmore. Quantification of artistic style through sparse coding analysis in the drawings of Pieter Bruegel the Elder. *Proceedings of the National Academy of Science, TODO USA*, 107(4):1279–1283, 2009.

R. Jenatton, J-Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. Technical report, 2009. preprint arXiv:0904.3523v1.

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)*, 2010a.

R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for hierarchical sparse coding. Technical report, 2010b. submitted, arXiv:1009.3139.

D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, 2001.

H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19, pages 801–808. MIT Press, Cambridge, MA, 2007.

References VI

M. Leordeanu, M. Hebert, and R. Sukthankar. Beyond local appearance: Category recognition from pairwise interactions of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2):337–365, 2000.

J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008a.

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

J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce. Discriminative sparse image models for class-specific edge detection and image interpretation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2008c.

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

References VII

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 IEEE International Conference on Computer Vision (ICCV)*, 2009b.

S. Mallat. *A Wavelet Tour of Signal Processing, Second Edition*. Academic Press, New York, September 1999.

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

H. M. Markowitz. The optimization of a quadratic function subject to linear constraints. *Naval Research Logistics Quarterly*, 3:111–133, 1956.

N. Meinshausen and P. Buehlmann. Stability selection. *Journal of the Royal Statistical Society, Series B*, 2010. to appear.

Y. Nesterov. A method for solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Math. Dokl.*, 27:372–376, 1983.

Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, CORE, 2007.

References VIII

J. Nocedal and SJ Wright. *Numerical Optimization*. Springer: New York, 2006. 2nd Edition.

B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37:3311–3325, 1997.

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, 2000.

D. Paliy, V. Katkovnik, R. Bilcu, S. Alenius, and K. Egiazarian. Spatially adaptive color filter array interpolation for noiseless and noisy data. *Intern. J. of Imaging Sys. and Tech.*, 17(3), 2007.

J. Portilla, V. Strela, MJ Wainwright, and EP Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, 2003.

M. Protter and M. Elad. Image sequence denoising via sparse and redundant representations. *IEEE Transactions on Image Processing*, 18(1):27–36, 2009.

S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

References IX

V. Roth and B. Fischer. The group-lasso for generalized linear models: uniqueness of solutions and efficient algorithms. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

P. Sprechmann, I. Ramirez, G. Sapiro, and Y. C. Eldar. Collaborative hierarchical sparse modeling. Technical report, 2010a. Preprint arXiv:1003.0400v1.

P. Sprechmann, I. Ramirez, G. Sapiro, and Y. C. Eldar. Collaborative hierarchical sparse modeling. Technical report, 2010b. Preprint arXiv:1003.0400v1.

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

S. Weisberg. *Applied Linear Regression*. Wiley, New York, 1980.

J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2005.

J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

References X

J. Yang, K. Yu, , and T. Huang. Supervised translation-invariant sparse coding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

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.

L. Zhang and X. Wu. Color demosaicking via directional linear minimum mean square-error estimation. *IEEE Transactions on Image Processing*, 14(12): 2167–2178, 2005.