

Part 1: Applications of Sparse Optimization

Mário A. T. Figueiredo¹ and Stephen J. Wright²

¹Instituto de Telecomunicações,
Instituto Superior Técnico, Lisboa, Portugal

²Computer Sciences Department,
University of Wisconsin,
Madison, WI, USA

ICCOPT, Lisbon, Portugal, July 2013

Many **inference** problems are formulated as **optimization** problems:

- image reconstruction
- image restoration/denoising
- supervised learning
- unsupervised learning
- statistical inference
- ...

Inference via Optimization

Many **inference** problems are formulated as **optimization** problems:

- image reconstruction
- image restoration/denoising
- supervised learning
- unsupervised learning
- statistical inference
- ...

Standard formulation:

- observed data: y
- unknown object (signal, image, vector, matrix,...): x
- inference criterion:

$$\hat{x} \in \arg \min_x g(x, y)$$

Inference criterion: $\hat{x} \in \arg \min_x g(x, y)$

Question 1: how to build g ? Where does it come from?

Answer: from the application domain (machine learning, signal processing, inverse problems, statistics, bioinformatics,...); we will see examples ahead.

Inference via Optimization

Inference criterion: $\hat{x} \in \arg \min_x g(x, y)$

Question 1: how to build g ? Where does it come from?

Answer: from the application domain (machine learning, signal processing, inverse problems, statistics, bioinformatics,...); we will see examples ahead.

Question 2: how to solve the optimization problem?

Answer: the focus of this tutorial.

Regularized Optimization

Inference criterion: $\hat{x} \in \arg \min_x g(x, y)$

Typical structure of g : $g(x, y) = h(x, y) + \tau\psi(x)$

Regularized Optimization

Inference criterion: $\hat{x} \in \arg \min_x g(x, y)$

Typical structure of g : $g(x, y) = h(x, y) + \tau\psi(x)$

- $h(x, y) \rightarrow$ how well x “fits” / “explains” the data y ;
(data term, log-likelihood, loss function, observation model,...)

Regularized Optimization

Inference criterion: $\hat{x} \in \arg \min_x g(x, y)$

Typical structure of g : $g(x, y) = h(x, y) + \tau\psi(x)$

- $h(x, y) \rightarrow$ how well x “fits” / “explains” the data y ;
(data term, log-likelihood, loss function, observation model,...)
- $\psi(x) \rightarrow$ knowledge/constraints/structure: the **regularizer**
- $\tau \geq 0$: the **regularization parameter** (or constant).

Regularized Optimization

Inference criterion: $\hat{x} \in \arg \min_x g(x, y)$

Typical structure of g : $g(x, y) = h(x, y) + \tau\psi(x)$

- $h(x, y) \rightarrow$ how well x “fits” / “explains” the data y ;
(data term, log-likelihood, loss function, observation model,...)
- $\psi(x) \rightarrow$ knowledge/constraints/structure: the **regularizer**
- $\tau \geq 0$: the **regularization parameter** (or constant).
- Since y is fixed, we often write simply $f(x) = h(x, y)$,

$$\min_x f(x) + \tau\psi(x)$$

Regularizers

Inference criterion:
$$\min_x f(x) + \tau\psi(x)$$

Typically, the unknown is a **vector** $x \in \mathbb{R}^n$
or a **matrix** $x \in \mathbb{R}^{n \times m}$

Inference criterion:
$$\min_x f(x) + \tau\psi(x)$$

Typically, the unknown is a **vector** $x \in \mathbb{R}^n$
or a **matrix** $x \in \mathbb{R}^{n \times m}$

Common **regularizers** impose/encourage one (or a combination of) the following characteristics:

- small norm (vector or matrix)
- sparsity (few nonzeros)
- specific nonzero patterns (e.g., group/tree structure)
- low-rank (matrix)
- smoothness or piece-wise smoothness

Unconstrained vs Constrained Formulations

- Tikhonov regularization: $\min_x f(x) + \tau\psi(x)$

Unconstrained vs Constrained Formulations

- Tikhonov regularization:

$$\min_x f(x) + \tau\psi(x)$$

- Morozov regularization:

$$\begin{array}{ll} \min_x & \psi(x) \\ \text{subject to} & f(x) \leq \varepsilon \end{array}$$

- Ivanov regularization:

$$\begin{array}{ll} \min_x & f(x) \\ \text{subject to} & \psi(x) \leq \delta \end{array}$$

Unconstrained vs Constrained Formulations

- Tikhonov regularization:
$$\min_x f(x) + \tau\psi(x)$$
- Morozov regularization:
$$\begin{array}{ll} \min_x & \psi(x) \\ \text{subject to} & f(x) \leq \varepsilon \end{array}$$
- Ivanov regularization:
$$\begin{array}{ll} \min_x & f(x) \\ \text{subject to} & \psi(x) \leq \delta \end{array}$$

Under mild conditions, these are all “*equivalent*”.

Morozov and Ivanov can be written as Tikhonov using indicator functions (more later).

Which one is more convenient is problem-dependent.

Example: Under- and Over-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

Example: Under- and Over-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Trivial case, A is invertible: $x = A^{-1}y$

Example: Under- and Over-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Trivial case, A is invertible: $x = A^{-1}y$
- Over-determined system ($m > n$); least squares solution ($\text{rank}(A) = n$):

$$\hat{x} = \arg \min_x \sum_{i=1}^n (y_i - (Ax)_i)^2 = \arg \min_x \|y - Ax\|_2^2 = (A^T A)^{-1} A^T y$$

Example: Under- and Over-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Trivial case, A is invertible: $x = A^{-1}y$
- Over-determined system ($m > n$); least squares solution ($\text{rank}(A) = n$):

$$\hat{x} = \arg \min_x \sum_{i=1}^n (y_i - (Ax)_i)^2 = \arg \min_x \|y - Ax\|_2^2 = (A^T A)^{-1} A^T y$$

- Under-determined system ($m < n$); minimum norm solution ($\text{rank}(A) = m$):

$$\hat{x} = \left\{ \begin{array}{l} \arg \min_x \|x\|_2^2 \\ \text{s.t. } Ax = y \end{array} \right\} = A^T (AA^T)^{-1} y$$

Example: Under- and Over-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Trivial case, A is invertible: $x = A^{-1}y$
- Over-determined system ($m > n$); least squares solution ($\text{rank}(A) = n$):

$$\hat{x} = \arg \min_x \sum_{i=1}^n (y_i - (Ax)_i)^2 = \arg \min_x \|y - Ax\|_2^2 = (A^T A)^{-1} A^T y$$

- Under-determined system ($m < n$); minimum norm solution ($\text{rank}(A) = m$):

$$\hat{x} = \left\{ \begin{array}{l} \arg \min_x \|x\|_2^2 \\ \text{s.t. } Ax = y \end{array} \right\} = A^T (AA^T)^{-1} y$$

- Non-trivial cases: resort to optimization and regularization.

Example: Under- and Over-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Trivial case, A is invertible: $x = A^{-1}y$
- Over-determined system ($m > n$); least squares solution ($\text{rank}(A) = n$):

$$\hat{x} = \arg \min_x \sum_{i=1}^n (y_i - (Ax)_i)^2 = \arg \min_x \|y - Ax\|_2^2 = (A^T A)^{-1} A^T y$$

- Under-determined system ($m < n$); minimum norm solution ($\text{rank}(A) = m$):

$$\hat{x} = \left\{ \begin{array}{l} \arg \min_x \|x\|_2^2 \\ \text{s.t. } Ax = y \end{array} \right\} = A^T (AA^T)^{-1} y$$

- Non-trivial cases: resort to optimization and regularization.
- Quadratic (Euclidean) losses and regularizers have a long and rich history: Gauss, Legendre, Wiener, Moore-Penrose, Tikhonov, ...

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (**homogeneity**);

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (homogeneity);
- $\|x + x'\| \leq \|x\| + \|x'\|$, for any $x, x' \in \mathcal{V}$ (triangle inequality);

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (homogeneity);
- $\|x + x'\| \leq \|x\| + \|x'\|$, for any $x, x' \in \mathcal{V}$ (triangle inequality);
- $\|x\| = 0 \Rightarrow x = 0$.

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (**homogeneity**);
- $\|x + x'\| \leq \|x\| + \|x'\|$, for any $x, x' \in \mathcal{V}$ (**triangle inequality**);
- $\|x\| = 0 \Rightarrow x = 0$.

Examples:

- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_p = \left(\sum_i |x_i|^p \right)^{1/p}$ (called **ℓ_p norm**, for $p \geq 1$).

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (**homogeneity**);
- $\|x + x'\| \leq \|x\| + \|x'\|$, for any $x, x' \in \mathcal{V}$ (**triangle inequality**);
- $\|x\| = 0 \Rightarrow x = 0$.

Examples:

- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_p = \left(\sum_i |x_i|^p \right)^{1/p}$ (called **ℓ_p norm**, for $p \geq 1$).
- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_\infty = \lim_{p \rightarrow \infty} \|x\|_p = \max\{|x_1|, \dots, |x_n|\}$

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (**homogeneity**);
- $\|x + x'\| \leq \|x\| + \|x'\|$, for any $x, x' \in \mathcal{V}$ (**triangle inequality**);
- $\|x\| = 0 \Rightarrow x = 0$.

Examples:

- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_p = \left(\sum_i |x_i|^p\right)^{1/p}$ (called **ℓ_p norm**, for $p \geq 1$).
- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_\infty = \lim_{p \rightarrow \infty} \|x\|_p = \max\{|x_1|, \dots, |x_n|\}$
- $\mathcal{V} = \mathbb{R}^{n \times n}$, $\|X\|_* = \text{trace}(\sqrt{X^T X})$ (matrix **nuclear norm**)

Norms: A Quick Review

Consider some real vector space \mathcal{V} , for example, \mathbb{R}^n or $\mathbb{R}^{n \times n}$, ...

Some function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ is a **norm** if it satisfies:

- $\|\alpha x\| = |\alpha| \|x\|$, for any $x \in \mathcal{V}$ and $\alpha \in \mathbb{R}$ (**homogeneity**);
- $\|x + x'\| \leq \|x\| + \|x'\|$, for any $x, x' \in \mathcal{V}$ (**triangle inequality**);
- $\|x\| = 0 \Rightarrow x = 0$.

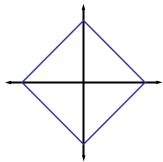
Examples:

- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_p = \left(\sum_i |x_i|^p\right)^{1/p}$ (called **ℓ_p norm**, for $p \geq 1$).
- $\mathcal{V} = \mathbb{R}^n$, $\|x\|_\infty = \lim_{p \rightarrow \infty} \|x\|_p = \max\{|x_1|, \dots, |x_n|\}$
- $\mathcal{V} = \mathbb{R}^{n \times n}$, $\|X\|_* = \text{trace}(\sqrt{X^T X})$ (matrix **nuclear norm**)

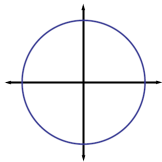
Also important (but not a norm): $\|x\|_0 = \lim_{p \rightarrow 0} \|x\|_p^p = |\{i : x_i \neq 0\}|$

Norm balls

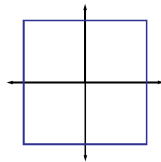
Radius r ball in ℓ_p norm: $B_p(r) = \{x \in \mathbb{R}^n : \|x\|_p \leq r\}$



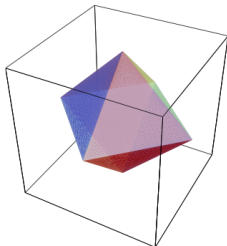
$p = 1$



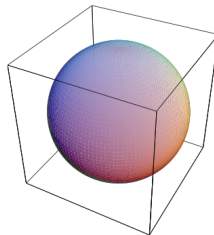
$p = 2$



$p = \infty$



$p = 1$



$p = 2$

Examples: Back to Under-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Under-determined system ($m < n$); minimum norm solution:

$$\hat{x} = \left\{ \begin{array}{l} \arg \min_x \|x\|_2^2 \\ \text{s.t. } Ax = y \end{array} \right\} = A^*(AA^*)^{-1}y$$

Examples: Back to Under-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Under-determined system ($m < n$); minimum norm solution:

$$\hat{x} = \left\{ \begin{array}{l} \arg \min_x \|x\|_2^2 \\ \text{s.t. } Ax = y \end{array} \right\} = A^*(AA^*)^{-1}y \neq x \text{ (in general)}$$

Examples: Back to Under-Constrained Systems

A simple linear inverse problem: from $y = Ax$, find x ($A \in \mathbb{R}^{m \times n}$)

- Under-determined system ($m < n$); minimum norm solution:

$$\hat{x} = \left\{ \begin{array}{l} \arg \min_x \|x\|_2^2 \\ \text{s.t. } Ax = y \end{array} \right\} = A^*(AA^*)^{-1}y \neq x \text{ (in general)}$$

- Can we hope to recover x ? **Yes!** ...if x is sparse enough ($\|x\|_0 < k$) and A satisfies some **conditions**, using

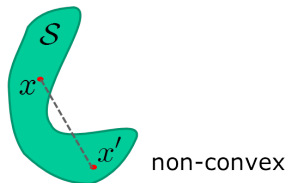
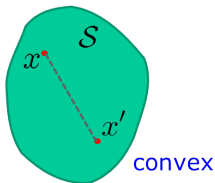
$$\hat{x} = \begin{array}{l} \arg \min_x \|x\|_0 \\ \text{s.t. } Ax = y \end{array}$$

Several proofs, under different conditions (more later).

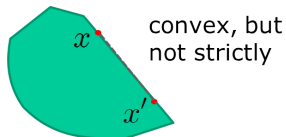
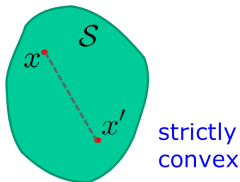
But, this is a **hard problem!** ℓ_0 “norm” is not **convex**.

Convex and strictly convex sets

\mathcal{S} is **convex** if $x, x' \in \mathcal{S} \Rightarrow \forall \lambda \in [0, 1], \lambda x + (1 - \lambda)x' \in \mathcal{S}$



\mathcal{S} is **strictly convex** if $x, x' \in \mathcal{S} \Rightarrow \forall \lambda \in (0, 1), \lambda x + (1 - \lambda)x' \in \text{int}(\mathcal{S})$



Review of Basics: Convex Functions

Extended real valued function: $f : \mathbb{R}^N \rightarrow \bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$

Domain: $\text{dom}(f) = \{x : f(x) \neq +\infty\}$

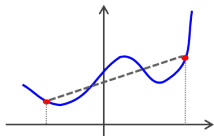
f is **proper** if $\text{dom}(f) \neq \emptyset$

f is **convex** if

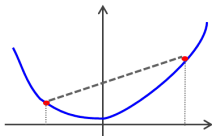
$$\forall \lambda \in [0, 1], x, x' \in \text{dom}(f) \quad f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x')$$

f is **strictly convex** if

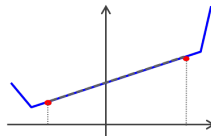
$$\forall \lambda \in (0, 1), x, x' \in \text{dom}(f) \quad f(\lambda x + (1 - \lambda)x') < \lambda f(x) + (1 - \lambda)f(x')$$



non-convex



strictly convex



convex, not strictly

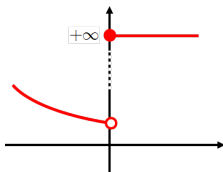
Lower Semi-Continuity: Why Is It Important?

A function $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ is lower semi-continuous (l.s.c.) if

$$\liminf_{x \rightarrow x_0} f(x) \geq f(x_0), \text{ for any } x_0 \in \text{dom}(f)$$

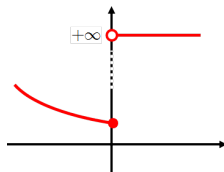
or, equivalently, $\{x : f(x) \leq \alpha\}$ is a closed set, for any $\alpha \in \mathbb{R}$

$$f(x) = \begin{cases} e^{-x}, & \text{if } x < 0 \\ +\infty, & \text{if } x \geq 0 \end{cases}$$



$$\text{dom}(f) =]-\infty, 0[, \arg \min_x f(x) = \emptyset$$

$$f(x) = \begin{cases} e^{-x}, & \text{if } x \leq 0 \\ +\infty, & \text{if } x > 0 \end{cases}$$



$$\text{dom}(f) =]-\infty, 0], \arg \min_x f(x) = \{0\}$$

Unless stated otherwise, we only consider l.s.c. functions.

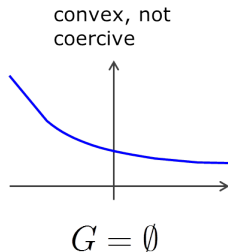
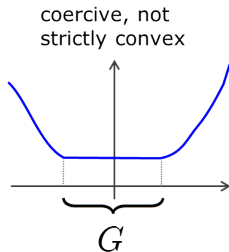
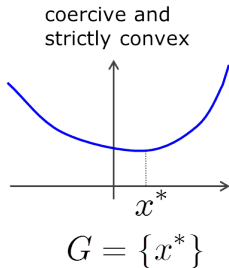
Coercivity, Convexity, and Minima

$$f : \mathbb{R}^N \rightarrow \bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$$

f is **coercive** if $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$

if f is **coercive**, then $G \equiv \arg \min_x f(x)$ is a non-empty set

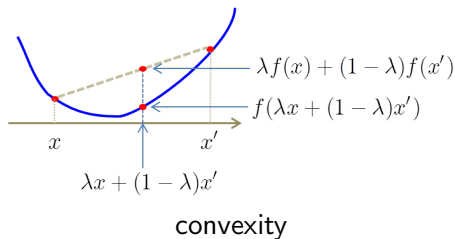
if f is **strictly convex**, then G has at most one element



Another Important Concept: Strong Convexity

Recall the definition of convex function: $\forall \lambda \in [0, 1]$,

$$f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x')$$



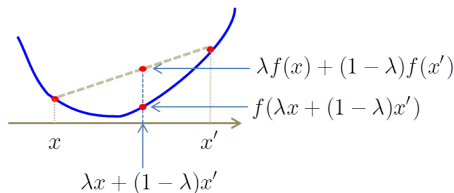
Another Important Concept: Strong Convexity

Recall the definition of convex function: $\forall \lambda \in [0, 1]$,

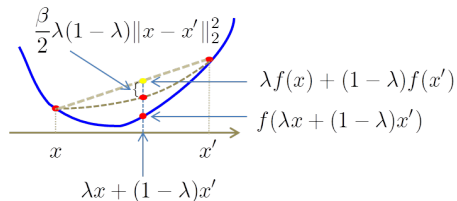
$$f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x')$$

A β -strongly convex function satisfies a stronger condition: $\forall \lambda \in [0, 1]$

$$f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x') - \frac{\beta}{2}\lambda(1 - \lambda)\|x - x'\|_2^2$$



convexity



strong convexity

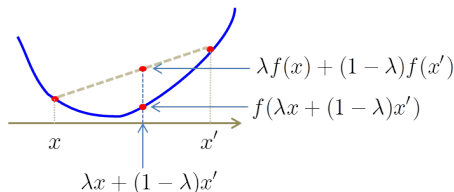
Another Important Concept: Strong Convexity

Recall the definition of convex function: $\forall \lambda \in [0, 1]$,

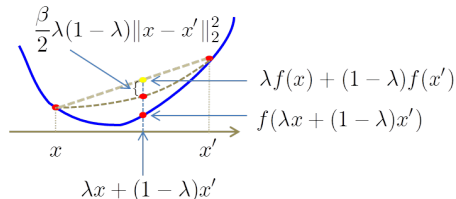
$$f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x')$$

A β -strongly convex function satisfies a stronger condition: $\forall \lambda \in [0, 1]$

$$f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x') - \frac{\beta}{2}\lambda(1 - \lambda)\|x - x'\|_2^2$$



convexity



strong convexity

Strong convexity \Rightarrow strict convexity.
 \nRightarrow

A Little More on Convex Functions

Let $f_1, \dots, f_N : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ be convex functions. Then

A Little More on Convex Functions

Let $f_1, \dots, f_N : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ be convex functions. Then

- $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $f(x) = \max\{f_1(x), \dots, f_N(x)\}$, is **convex**.

A Little More on Convex Functions

Let $f_1, \dots, f_N : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ be convex functions. Then

- $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $f(x) = \max\{f_1(x), \dots, f_N(x)\}$, is **convex**.
- $g : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $g(x) = f_1(L(x))$, where L is **affine**, is **convex**.

Note: L is affine $\Leftrightarrow L(x) - L(0)$ is linear; e.g. $L(x) = Ax + b$.

A Little More on Convex Functions

Let $f_1, \dots, f_N : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ be convex functions. Then

- $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $f(x) = \max\{f_1(x), \dots, f_N(x)\}$, is **convex**.
- $g : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $g(x) = f_1(L(x))$, where L is **affine**, is **convex**.
Note: L is affine $\Leftrightarrow L(x) - L(0)$ is linear; e.g. $L(x) = Ax + b$.
- $h : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $h(x) = \sum_{j=1}^N \alpha_j f_j(x)$, for $\alpha_j > 0$, is **convex**.

A Little More on Convex Functions

Let $f_1, \dots, f_N : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ be convex functions. Then

- $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $f(x) = \max\{f_1(x), \dots, f_N(x)\}$, is **convex**.
- $g : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $g(x) = f_1(L(x))$, where L is **affine**, is **convex**.
Note: L is affine $\Leftrightarrow L(x) - L(0)$ is linear; e.g. $L(x) = Ax + b$.
- $h : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, defined as $h(x) = \sum_{j=1}^N \alpha_j f_j(x)$, for $\alpha_j > 0$, is **convex**.

An important function: the **indicator** of a set $C \subset \mathbb{R}^n$,

$$\iota_C : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}, \quad \iota_C(x) = \begin{cases} 0 & \Leftarrow x \in C \\ +\infty & \Leftarrow x \notin C \end{cases}$$

If C is a **closed convex set**, ι_C is a **l.s.c. convex function**.

The Case of Differentiable Functions

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be twice differentiable and consider its **Hessian** matrix at x , denoted $\nabla^2 f(x)$ (or $Hf(x)$):

$$(\nabla^2 f(x))_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}, \text{ for } i, j = 1, \dots, n.$$

The Case of Differentiable Functions

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be twice differentiable and consider its **Hessian** matrix at x , denoted $\nabla^2 f(x)$ (or $Hf(x)$):

$$(\nabla^2 f(x))_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}, \quad \text{for } i, j = 1, \dots, n.$$

- f is **convex** \Leftrightarrow its Hessian $\nabla^2 f(x)$ is positive semidefinite \forall_x

The Case of Differentiable Functions

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be twice differentiable and consider its **Hessian** matrix at x , denoted $\nabla^2 f(x)$ (or $Hf(x)$):

$$(\nabla^2 f(x))_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}, \text{ for } i, j = 1, \dots, n.$$

- f is **convex** \Leftrightarrow its Hessian $\nabla^2 f(x)$ is positive semidefinite \forall_x
- f is **strictly convex** \Leftarrow its Hessian $\nabla^2 f(x)$ is positive definite \forall_x

The Case of Differentiable Functions

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be twice differentiable and consider its **Hessian** matrix at x , denoted $\nabla^2 f(x)$ (or $Hf(x)$):

$$(\nabla^2 f(x))_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}, \text{ for } i, j = 1, \dots, n.$$

- f is **convex** \Leftrightarrow its Hessian $\nabla^2 f(x)$ is positive semidefinite \forall_x
- f is **strictly convex** \Leftarrow its Hessian $\nabla^2 f(x)$ is positive definite \forall_x
- f is **β -strongly convex** \Leftrightarrow its Hessian $\nabla^2 f(x) \succeq \beta I$, with $\beta > 0$, \forall_x .

More on the Relationship Between ℓ_1 and ℓ_0

Finding the sparsest solution is NP-hard (Muthukrishnan, 2005).

$$\begin{aligned}\hat{w} &= \arg \min_w \|w\|_0 \\ \text{s. t. } &\|Aw - y\|_2^2 \leq \delta\end{aligned}$$

More on the Relationship Between ℓ_1 and ℓ_0

Finding the sparsest solution is NP-hard (Muthukrishnan, 2005).

$$\begin{aligned}\hat{w} &= \arg \min_w \|w\|_0 \\ \text{s. t. } &\|Aw - y\|_2^2 \leq \delta\end{aligned}$$

The related best subset selection problem is also NP-hard (Amaldi and Kann, 1998; Davis et al., 1997).

$$\begin{aligned}\hat{w} &= \arg \min_w \|Aw - y\|_2^2 \\ \text{s. t. } &\|w\|_0 \leq \tau\end{aligned}$$

More on the Relationship Between ℓ_1 and ℓ_0

Finding the sparsest solution is NP-hard (Muthukrishnan, 2005).

$$\begin{aligned}\hat{w} &= \arg \min_w \|w\|_0 \\ \text{s. t. } &\|Aw - y\|_2^2 \leq \delta\end{aligned}$$

The related best subset selection problem is also NP-hard (Amaldi and Kann, 1998; Davis et al., 1997).

$$\begin{aligned}\hat{w} &= \arg \min_w \|Aw - y\|_2^2 \\ \text{s. t. } &\|w\|_0 \leq \tau\end{aligned}$$

Under conditions, replacing ℓ_0 with ℓ_1 yields “similar” results:
central issue in compressive sensing (CS) (Candès et al., 2006a; Donoho, 2006)

Compressive Sensing in a Nutshell

The diagram illustrates the compressive sensing equation $y = Aw$. On the left, a vertical vector y of size $N \times 1$ is shown. In the center, a matrix A of size $N \times D$ is shown, with the condition $N \ll D$ indicated below it. On the right, a vertical vector w of size $D \times 1$ is shown, with the text "(+ noise)" next to it. The vector w is composed of colored blocks, and the matrix A is a sparse matrix with colored blocks.

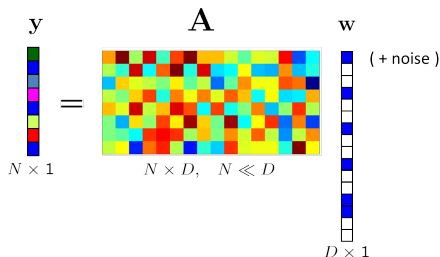
Even in the noiseless case, it seems impossible to recover w from y

Compressive Sensing in a Nutshell

The diagram illustrates the compressive sensing equation $y = Aw$. On the left, a vertical vector y of size $N \times 1$ is shown with a colorful, noisy pattern. In the center, a matrix A of size $N \times D$ is shown, where $N \ll D$. The matrix A is a random measurement matrix with a colorful, noisy pattern. On the right, a vertical vector w of size $D \times 1$ is shown, with most elements being white (zero) and a few blue squares (non-zero), indicating sparsity. The text "(+ noise)" is next to the vector w .

Even in the noiseless case, it seems impossible to recover w from y ...unless, w is **sparse** and A has some properties.

Compressive Sensing in a Nutshell



Even in the noiseless case, it seems impossible to recover w from y ...unless, w is **sparse** and A has some properties.

If w is sparse enough and A has certain properties, then w is stably recovered via (Haupt and Nowak, 2006)

$$\begin{aligned} \hat{w} &= \arg \min_w \|w\|_0 \\ \text{s. t. } \|Aw - y\| &\leq \delta \end{aligned}$$

NP-hard!

Compressive Sensing in a Nutshell

Under some conditions on A (e.g., the **restricted isometry property (RIP)**), ℓ_0 can be replaced with ℓ_1 (Candès et al., 2006b):

$$\begin{aligned}\hat{w} &= \arg \min_w \|w\|_1 \\ &\text{subject to } \|Aw - y\| \leq \delta\end{aligned}\quad \text{convex problem}$$

Compressive Sensing in a Nutshell

Under some conditions on A (e.g., the **restricted isometry property (RIP)**), ℓ_0 can be replaced with ℓ_1 (Candès et al., 2006b):

$$\begin{aligned} \hat{w} &= \arg \min_w \|w\|_1 \\ &\text{subject to } \|Aw - y\| \leq \delta \end{aligned} \quad \text{convex problem}$$

Matrix A satisfies the **RIP** of order k , with constant $\delta_k \in (0, 1)$, if

$$\|w\|_0 \leq k \Rightarrow (1 - \delta_k) \|w\|_2^2 \leq \|Aw\| \leq (1 + \delta_k) \|w\|_2^2$$

...i.e., for k -sparse vectors, A is approximately an isometry.

Compressive Sensing in a Nutshell

Under some conditions on A (e.g., the **restricted isometry property (RIP)**), ℓ_0 can be replaced with ℓ_1 (Candès et al., 2006b):

$$\begin{aligned}\hat{w} &= \arg \min_w \|w\|_1 \\ &\text{subject to } \|Aw - y\| \leq \delta\end{aligned}\quad \text{convex problem}$$

Matrix A satisfies the **RIP** of order k , with constant $\delta_k \in (0, 1)$, if

$$\|w\|_0 \leq k \Rightarrow (1 - \delta_k) \|w\|_2^2 \leq \|Aw\| \leq (1 + \delta_k) \|w\|_2^2$$

...i.e., for k -sparse vectors, A is approximately an isometry.

Other properties (**spark** and **null space property (NSP)**) can be used; caveat: checking **RIP**, **NSP**, **spark** is **NP-hard** (Tillmann and Pfetsch, 2012).

Examples: Back to Under-Constrained Systems

Let \bar{x} be the **sparsest solution** of $Ax = y$, where $A \in \mathbb{R}^{m \times n}$ and $m < n$.

$$\bar{x} = \arg \min \|x\|_0 \text{ s.t. } Ax = y.$$

Consider the ℓ_1 norm version: $\min_x \|x\|_1 \text{ s.t. } Ax = y$

Examples: Back to Under-Constrained Systems

Let \bar{x} be the **sparsest solution** of $Ax = y$, where $A \in \mathbb{R}^{m \times n}$ and $m < n$.

$$\bar{x} = \arg \min \|x\|_0 \text{ s.t. } Ax = y.$$

Consider the ℓ_1 norm version: $\min_x \|x\|_1 \text{ s.t. } Ax = y$

Advantage: this is a convex problem! Fact: **all norms are convex**.

Examples: Back to Under-Constrained Systems

Let \bar{x} be the **sparsest solution** of $Ax = y$, where $A \in \mathbb{R}^{m \times n}$ and $m < n$.

$$\bar{x} = \arg \min \|x\|_0 \text{ s.t. } Ax = y.$$

Consider the ℓ_1 norm version: $\min_x \|x\|_1 \text{ s.t. } Ax = y$

Advantage: this is a convex problem! Fact: **all norms are convex**.

Of course, \bar{x} solves this problem too, if $\|\bar{x} + v\|_1 \geq \|\bar{x}\|_1, \forall v \in \ker(A)$.

Recall: $\ker(A) = \{x \in \mathbb{R}^n : Ax = 0\}$ is the **kernel** (a.k.a. **null space**) of A .

Examples: Back to Under-Constrained Systems

Let \bar{x} be the **sparsest solution** of $Ax = y$, where $A \in \mathbb{R}^{m \times n}$ and $m < n$.

$$\bar{x} = \arg \min \|x\|_0 \text{ s.t. } Ax = y.$$

Consider the ℓ_1 norm version: $\min_x \|x\|_1 \text{ s.t. } Ax = y$

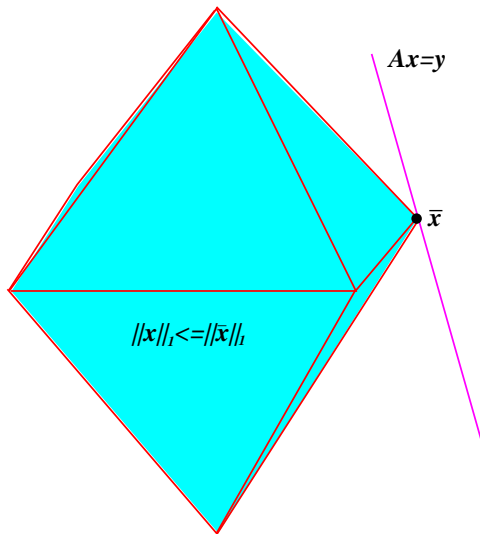
Advantage: this is a convex problem! Fact: **all norms are convex**.

Of course, \bar{x} solves this problem too, if $\|\bar{x} + v\|_1 \geq \|\bar{x}\|_1, \forall v \in \ker(A)$.

Recall: $\ker(A) = \{x \in \mathbb{R}^n : Ax = 0\}$ is the **kernel** (a.k.a. **null space**) of A .

Next: elementary analysis by Yin and Zhang (2008), based on work by Kashin (1977) and Garnaev and Gluskin (1984).

Sparse Reconstruction with ℓ_1 Optimization



Equivalence Between ℓ_1 and ℓ_0 Optimization

- Minimum ℓ_0 (sparsest) solution: $\bar{x} \in \arg \min \|x\|_0$ s.t. $Ax = y$.
- Minimum ℓ_1 solution(s): $G = \arg \min \|x\|_1$ s.t. $Ax = y$.

Equivalence Between ℓ_1 and ℓ_0 Optimization

- Minimum ℓ_0 (sparsest) solution: $\bar{x} \in \arg \min \|x\|_0$ s.t. $Ax = y$.
- Minimum ℓ_1 solution(s): $G = \arg \min \|x\|_1$ s.t. $Ax = y$.
- $\bar{x} \in G$, if $\|\bar{x} + v\|_1 \geq \|\bar{x}\|_1, \forall v \in \ker(A)$

Equivalence Between ℓ_1 and ℓ_0 Optimization

- Minimum ℓ_0 (sparsest) solution: $\bar{x} \in \arg \min \|x\|_0$ s.t. $Ax = y$.
- Minimum ℓ_1 solution(s): $G = \arg \min \|x\|_1$ s.t. $Ax = y$.
- $\bar{x} \in G$, if $\|\bar{x} + v\|_1 \geq \|\bar{x}\|_1, \forall v \in \ker(A)$
- Letting $S = \{i : \bar{x}_i \neq 0\}$ and $Z = \{1, \dots, n\} \setminus S$

$$\begin{aligned}\|\bar{x} + v\|_1 &= \|\bar{x}_S + v_S\|_1 + \|v_Z\|_1 \\ &\geq \|\bar{x}_S\|_1 + \|v_Z\|_1 - \|v_S\|_1 \\ &= \|\bar{x}\|_1 + \|v\|_1 - 2\|v_S\|_1 \\ &\geq \|\bar{x}\|_1 + \|v\|_1 - 2\sqrt{k}\|v\|_2.\end{aligned}$$

Equivalence Between ℓ_1 and ℓ_0 Optimization

- Minimum ℓ_0 (sparsest) solution: $\bar{x} \in \arg \min \|x\|_0$ s.t. $Ax = y$.
- Minimum ℓ_1 solution(s): $G = \arg \min \|x\|_1$ s.t. $Ax = y$.
- $\bar{x} \in G$, if $\|\bar{x} + v\|_1 \geq \|\bar{x}\|_1, \forall v \in \ker(A)$
- Letting $S = \{i : \bar{x}_i \neq 0\}$ and $Z = \{1, \dots, n\} \setminus S$

$$\begin{aligned}\|\bar{x} + v\|_1 &= \|\bar{x}_S + v_S\|_1 + \|v_Z\|_1 \\ &\geq \|\bar{x}_S\|_1 + \|v_Z\|_1 - \|v_S\|_1 \\ &= \|\bar{x}\|_1 + \|v\|_1 - 2\|v_S\|_1 \\ &\geq \|\bar{x}\|_1 + \|v\|_1 - 2\sqrt{k}\|v\|_2.\end{aligned}$$

Hence, $\bar{x} \in G$, if $\frac{1}{2} \frac{\|v\|_1}{\|v\|_2} \geq \sqrt{k}, \forall v \in \ker(A)$

...but, in general, we have only: $1 \leq \frac{\|v\|_1}{\|v\|_2} \leq \sqrt{n}$

Equivalence Between ℓ_1 and ℓ_0 Optimization

- Minimum ℓ_0 (sparsest) solution: $\bar{x} \in \arg \min \|x\|_0$ s.t. $Ax = y$.
- Minimum ℓ_1 solution(s): $G = \arg \min \|x\|_1$ s.t. $Ax = y$.
- $\bar{x} \in G$, if $\|\bar{x} + v\|_1 \geq \|\bar{x}\|_1, \forall v \in \ker(A)$
- Letting $S = \{i : \bar{x}_i \neq 0\}$ and $Z = \{1, \dots, n\} \setminus S$

$$\begin{aligned}\|\bar{x} + v\|_1 &= \|\bar{x}_S + v_S\|_1 + \|v_Z\|_1 \\ &\geq \|\bar{x}_S\|_1 + \|v_Z\|_1 - \|v_S\|_1 \\ &= \|\bar{x}\|_1 + \|v\|_1 - 2\|v_S\|_1 \\ &\geq \|\bar{x}\|_1 + \|v\|_1 - 2\sqrt{k}\|v\|_2.\end{aligned}$$

Hence, $\bar{x} \in G$, if $\frac{1}{2} \frac{\|v\|_1}{\|v\|_2} \geq \sqrt{k}, \forall v \in \ker(A)$

...but, in general, we have only: $1 \leq \frac{\|v\|_1}{\|v\|_2} \leq \sqrt{n}$

However, we may have $\frac{\|v\|_1}{\|v\|_2} \gg 1$, if v is restricted to a random subspace.

Bounding the ℓ_1/ℓ_2 Ratio in Random Matrices

If the elements of $A \in \mathbb{R}^{m \times n}$ are sampled i.i.d. from $\mathcal{N}(0, 1)$ (zero mean, unit variance Gaussian), then, with high probability,

$$\frac{\|v\|_1}{\|v\|_2} \geq \frac{C\sqrt{m}}{\sqrt{\log(n/m)}}, \quad \text{for all } v \in \ker(A),$$

for some constant C (based on concentration of measure phenomena).

Thus, with high probability, $\bar{x} \in G$, if

$$m \geq \frac{4}{C^2} k \log n \quad \Leftrightarrow \quad k \leq \frac{C^2}{4 \log n}$$

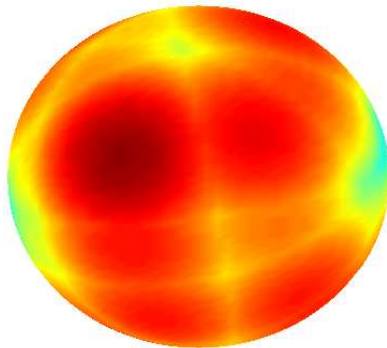
Conclusion: Can solve under-determined system, where A has i.i.d. $\mathcal{N}(0, 1)$ elements, by solving

$$\min_x \|x\|_1 \quad \text{s.t.} \quad Ax = b,$$

(a convex problem), if the solution is sparse enough

Ratio $\|v\|_1/\|v\|_2$ on Random Null Spaces

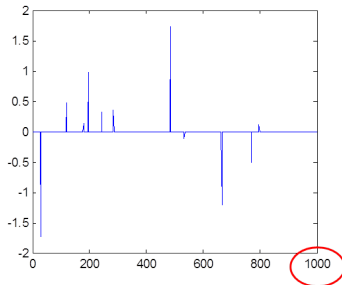
Random $A \in \mathbb{R}^{4 \times 7}$, showing ratio $\|v\|_1$ for $v \in \ker(A)$ with $\|v\|_2 = 1$



Blue: $\|v\|_1 \approx 1$. Red: ratio $\approx \sqrt{7}$. Note that $\|v\|_1$ is well away from the lower bound of 1 over the whole nullspace.

When Data is Noisy

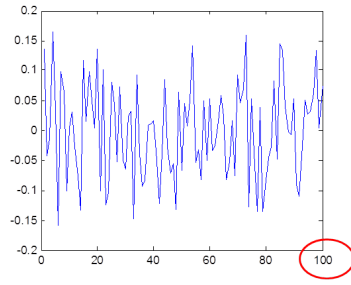
Sparse vector \mathbf{x}



$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$$

Random matrix

Observed data \mathbf{y}

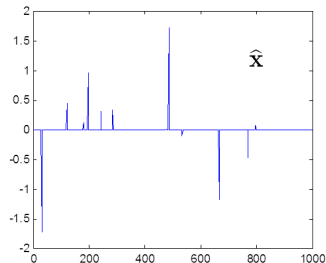


Under certain conditions, “perfect” recovery is possible

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left\{ \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + 2\lambda \|\mathbf{x}\|_1 \right\}$$

[Candès, Romberg, Tao, 2004 – 2006]

[Donoho, 2006]



The Ubiquitous ℓ_1 Norm

- Lasso (*least absolute shrinkage and selection operator*) (Tibshirani, 1996)
a.k.a. *basis pursuit denoising* (Chen et al., 1995):

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \tau \|x\|_1 \quad \text{or} \quad \min_x \|Ax - y\|_2^2 \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

or, more generally,

$$\min_x f(x) + \lambda \|x\|_1 \quad \text{or} \quad \min_x f(x) \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

The Ubiquitous ℓ_1 Norm

- Lasso (*least absolute shrinkage and selection operator*) (Tibshirani, 1996)
a.k.a. *basis pursuit denoising* (Chen et al., 1995):

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \tau \|x\|_1 \quad \text{or} \quad \min_x \|Ax - y\|_2^2 \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

or, more generally,

$$\min_x f(x) + \lambda \|x\|_1 \quad \text{or} \quad \min_x f(x) \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

- Widely used outside and much earlier than compressive sensing (statistics, signal processing, neural networks, ...).

- Lasso (*least absolute shrinkage and selection operator*)
a.k.a. *basis pursuit*

$$\min_x \frac{1}{2} \|Ax - b\|_2^2$$

or, more generally

$$\min_x$$

- Widely used in statistics, signal processing, and machine learning

- Geology/geophysics
 - Claerbout and Muir (1973)
 - Taylor et al. (1979)
 - Levy and Fullager (1981)
 - Oldenburg et al. (1983)
 - Santosa and Symes (1988)
- Radio astronomy
 - Högbom (1974)
 - Schwarz (1978)
- Fourier transform spectroscopy
 - Kawata et al. (1983)
 - Mammone (1983)
 - Minami et al. (1985)
- NMR spectroscopy
 - Barkhuijsen (1985)
 - Newman (1988)
- Medical ultrasound
 - Papoulis and Chamzas (1979)

from (Goyal et al, 2010)

(Tibshirani, 1996)

$$\text{s.t. } \|x\|_1 \leq \delta$$

$$\|x\|_1 \leq \delta$$

compressive sensing

The Ubiquitous ℓ_1 Norm

- Lasso (*least absolute shrinkage and selection operator*) (Tibshirani, 1996)
a.k.a. *basis pursuit denoising* (Chen et al., 1995):

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \tau \|x\|_1 \quad \text{or} \quad \min_x \|Ax - y\|_2^2 \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

or, more generally,

$$\min_x f(x) + \lambda \|x\|_1 \quad \text{or} \quad \min_x f(x) \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

- Widely used outside and much earlier than compressive sensing (statistics, signal processing, neural networks, ...).
- Many extensions: namely to express structured sparsity (more later).

The Ubiquitous ℓ_1 Norm

- Lasso (*least absolute shrinkage and selection operator*) (Tibshirani, 1996) a.k.a. *basis pursuit denoising* (Chen et al., 1995):

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \tau \|x\|_1 \quad \text{or} \quad \min_x \|Ax - y\|_2^2 \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

or, more generally,

$$\min_x f(x) + \lambda \|x\|_1 \quad \text{or} \quad \min_x f(x) \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

- Widely used outside and much earlier than compressive sensing (statistics, signal processing, neural networks, ...).
- Many extensions: namely to express structured sparsity (more later).
- Why does ℓ_1 yield sparse solutions? (next slides)

The Ubiquitous ℓ_1 Norm

- Lasso (*least absolute shrinkage and selection operator*) (Tibshirani, 1996) a.k.a. *basis pursuit denoising* (Chen et al., 1995):

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \tau \|x\|_1 \quad \text{or} \quad \min_x \|Ax - y\|_2^2 \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

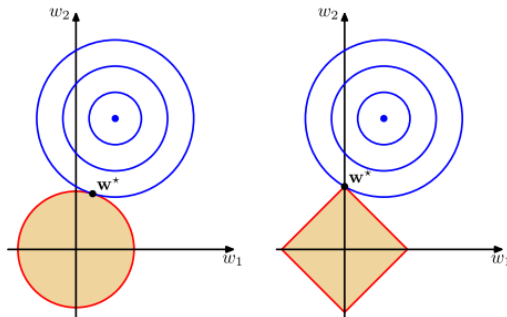
or, more generally,

$$\min_x f(x) + \lambda \|x\|_1 \quad \text{or} \quad \min_x f(x) \quad \text{s.t.} \quad \|x\|_1 \leq \delta$$

- Widely used outside and much earlier than compressive sensing (statistics, signal processing, neural networks, ...).
- Many extensions: namely to express structured sparsity (more later).
- Why does ℓ_1 yield sparse solutions? (next slides)
- How to solve these problems? (this tutorial)

Why ℓ_1 Yields Sparse Solution

$$w^* = \underset{\text{s.t. } \|w\|_2 \leq \delta}{\operatorname{arg\,min}_w} \|Aw - y\|_2^2 \quad \text{vs} \quad w^* = \underset{\text{s.t. } \|w\|_1 \leq \delta}{\operatorname{arg\,min}_w} \|Aw - y\|_2^2$$



Why ℓ_1 Yields Sparse Solution

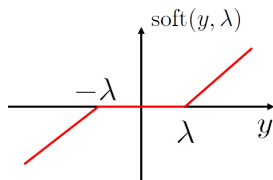
The simplest problem with ℓ_1 regularization

$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \lambda|w| = \text{soft}(y, \lambda) = \begin{cases} y - \lambda & \Leftarrow y > \lambda \\ 0 & \Leftarrow |y| \leq \lambda \\ y + \lambda & \Leftarrow y < -\lambda \end{cases}$$

Why ℓ_1 Yields Sparse Solution

The simplest problem with ℓ_1 regularization

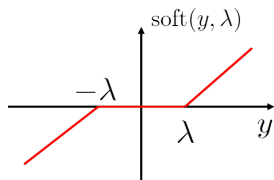
$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \lambda|w| = \text{soft}(y, \lambda) = \begin{cases} y - \lambda & \Leftarrow y > \lambda \\ 0 & \Leftarrow |y| \leq \lambda \\ y + \lambda & \Leftarrow y < -\lambda \end{cases}$$



Why ℓ_1 Yields Sparse Solution

The simplest problem with ℓ_1 regularization

$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \lambda|w| = \text{soft}(y, \lambda) = \begin{cases} y - \lambda & \Leftarrow y > \lambda \\ 0 & \Leftarrow |y| \leq \lambda \\ y + \lambda & \Leftarrow y < -\lambda \end{cases}$$

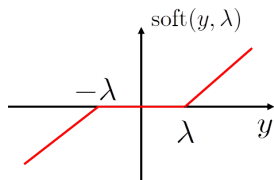


...by the way, how is this solved? (more later).

Why ℓ_1 Yields Sparse Solution

The simplest problem with ℓ_1 regularization

$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \lambda|w| = \text{soft}(y, \lambda) = \begin{cases} y - \lambda & \Leftarrow y > \lambda \\ 0 & \Leftarrow |y| \leq \lambda \\ y + \lambda & \Leftarrow y < -\lambda \end{cases}$$



...by the way, how is this solved? (more later).

Contrast with the squared ℓ_2 (ridge) regularizer (linear scaling):

$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \frac{\lambda}{2} w^2 = \frac{1}{1 + \lambda} y$$

More on the Relationship Between ℓ_1 and ℓ_0

The ℓ_0 “norm” (number of non-zeros): $\|w\|_0 = |\{i : w_i \neq 0\}|$.

Not a norm, not convex, but in the simple case...

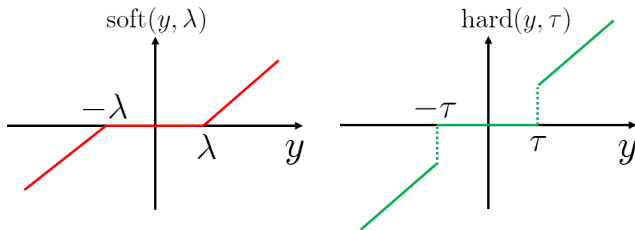
$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \lambda|w|_0 = \text{hard}(y, \sqrt{2\lambda}) = \begin{cases} y & \Leftarrow |y| > \sqrt{2\lambda} \\ 0 & \Leftarrow |y| \leq \sqrt{2\lambda} \end{cases}$$

More on the Relationship Between ℓ_1 and ℓ_0

The ℓ_0 “norm” (number of non-zeros): $\|w\|_0 = |\{i : w_i \neq 0\}|$.

Not a norm, not convex, but in the simple case...

$$\hat{w} = \arg \min_w \frac{1}{2}(w - y)^2 + \lambda|w|_0 = \text{hard}(y, \sqrt{2\lambda}) = \begin{cases} y & \Leftarrow |y| > \sqrt{2\lambda} \\ 0 & \Leftarrow |y| \leq \sqrt{2\lambda} \end{cases}$$



Another Application: Images

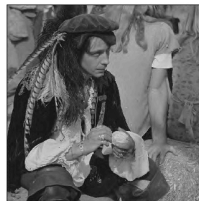
Natural images are well represented by a few coefficients in some bases.

- Images ($N \times M \equiv n$ pixels) are represented by vectors $x \in \mathbb{R}^n$

Another Application: Images

Natural images are well represented by a few coefficients in some bases.

- Images ($N \times M \equiv n$ pixels) are represented by vectors $x \in \mathbb{R}^n$
- Typical images have representations $x = Ww$ that are sparse ($\|w\|_0 \ll n$) on some bases ($W^T W = W W^T = I$), such as **wavelets**.

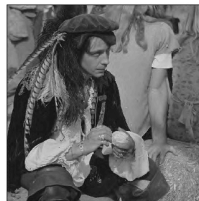


Original 1000×1000 image $x \in \mathbb{R}^{10^6}$...only its 25000 largest coefficients.

Another Application: Images

Natural images are well represented by a few coefficients in some bases.

- Images ($N \times M \equiv n$ pixels) are represented by vectors $x \in \mathbb{R}^n$
- Typical images have representations $x = Ww$ that are sparse ($\|w\|_0 \ll n$) on some bases ($W^T W = W W^T = I$), such as **wavelets**.



Original 1000×1000 image $x \in \mathbb{R}^{10^6}$...only its 25000 largest coefficients.

- Also (even more) true with an over-complete tight frame; W is “fat” (more columns than rows) and $W W^T = I$, but $W^T W \neq I$.

Application to Image Deblurring/Deconvolution

blurred



restored



$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{Ax} - \mathbf{y}\|_2^2 + \tau \|\mathbf{x}\|_1$$

$$\mathbf{A} = \mathbf{BW}$$

convolution (blur)

wavelet basis (or tight frame)

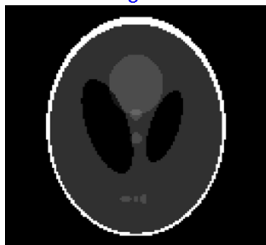
Application to Magnetic Resonance Imaging

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{Ax} - \mathbf{y}\|_2^2 + \tau \|\mathbf{x}\|_1$$

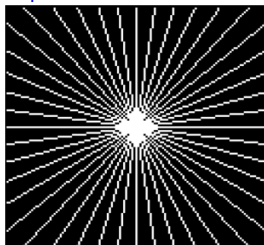
$$\mathbf{A} = \mathbf{MUW}$$

binary mask    wavelet basis (or tight frame)
discrete Fourier transform

original



acquired slices in DFT domain



reconstruction $\mathbf{W}\hat{\mathbf{x}}$



Machine/Statistical Learning: Linear Regression

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature/variable vectors) and $y_i \in \mathbb{R}$ (outputs).

Goal: find “good” linear function: $\hat{y} = \sum_{j=1}^d w_j x_j + w_{d+1} = [x^T \mathbf{1}] w$

Machine/Statistical Learning: Linear Regression

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature/variable vectors) and $y_i \in \mathbb{R}$ (outputs).

Goal: find “good” linear function: $\hat{y} = \sum_{j=1}^d w_j x_j + w_{d+1} = [x^T 1]w$

Assumption: data generated i.i.d. by some underlying distribution $P_{X,Y}$

Mean squared error: $\min_w \mathbb{E}(Y - [X^T 1]w)^2$ impossible! $P_{X,Y}$ unknown

Machine/Statistical Learning: Linear Regression

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature/variable vectors) and $y_i \in \mathbb{R}$ (outputs).

Goal: find “good” linear function: $\hat{y} = \sum_{j=1}^d w_j x_j + w_{d+1} = [x^T 1]w$

Assumption: data generated i.i.d. by some underlying distribution $P_{X,Y}$

Mean squared error: $\min_w \mathbb{E}(Y - [X^T 1]w)^2$ impossible! $P_{X,Y}$ unknown

Empirical error: $\min_w \frac{1}{N} \sum_{i=1}^N (y_i - [x_i^T 1]w)^2 = \min_w \frac{1}{N} \|y - Aw\|_2^2,$

design matrix: $A_{ij} = (x_i)_j$ (j -th component of i -th sample, $A_{i(d+1)} = 1$)

Machine/Statistical Learning: Linear Regression

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature/variable vectors) and $y_i \in \mathbb{R}$ (outputs).

Goal: find “good” linear function: $\hat{y} = \sum_{j=1}^d w_j x_j + w_{d+1} = [x^T 1]w$

Assumption: data generated i.i.d. by some underlying distribution $P_{X,Y}$

Mean squared error: $\min_w \mathbb{E}(Y - [X^T 1]w)^2$ impossible! $P_{X,Y}$ unknown

Empirical error: $\min_w \frac{1}{N} \sum_{i=1}^N (y_i - [x_i^T 1]w)^2 = \min_w \frac{1}{N} \|y - Aw\|_2^2,$

design matrix: $A_{ij} = (x_i)_j$ (j -th component of i -th sample, $A_{i(d+1)} = 1$)

Regularization: $\min_w \|y - Aw\|_2^2 + \tau\psi(w)$

Machine/Statistical Learning: Linear Classification

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature vectors) and $y_i \in \{-1, +1\}$ (labels).

Goal: find “good” linear classifier (i.e., find the optimal weights):

$$\hat{y} = \text{sign}([x^T \ 1]w) = \text{sign}\left(w_{d+1} + \sum_{j=1}^d w_j x_j\right)$$

Machine/Statistical Learning: Linear Classification

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature vectors) and $y_i \in \{-1, +1\}$ (labels).

Goal: find “good” linear classifier (i.e., find the optimal weights):

$$\hat{y} = \text{sign}([x^T 1]w) = \text{sign}\left(w_{d+1} + \sum_{j=1}^d w_j x_j\right)$$

Assumption: data generated i.i.d. by some underlying distribution $P_{X,Y}$

Expected error: $\min_{w \in \mathbb{R}^{d+1}} \mathbb{E}(1_{Y([X^T 1]w) < 0})$ impossible! $P_{X,Y}$ unknown

Machine/Statistical Learning: Linear Classification

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature vectors) and $y_i \in \{-1, +1\}$ (labels).

Goal: find “good” linear classifier (i.e., find the optimal weights):

$$\hat{y} = \text{sign}([x^T 1]w) = \text{sign}\left(w_{d+1} + \sum_{j=1}^d w_j x_j\right)$$

Assumption: data generated i.i.d. by some underlying distribution $P_{X,Y}$

Expected error: $\min_{w \in \mathbb{R}^{d+1}} \mathbb{E}(1_{Y([X^T 1]w) < 0})$ impossible! $P_{X,Y}$ unknown

Empirical error (EE): $\min_w \frac{1}{N} \sum_{i=1}^N h(\underbrace{y_i ([x^T 1]w)}_{\text{margin}})$, where $h(z) = 1_{z < 0}$.

Machine/Statistical Learning: Linear Classification

Data N pairs $(x_1, y_1), \dots, (x_N, y_N)$, where $x_i \in \mathbb{R}^d$ (feature vectors) and $y_i \in \{-1, +1\}$ (labels).

Goal: find “good” linear classifier (i.e., find the optimal weights):

$$\hat{y} = \text{sign}([x^T \ 1]w) = \text{sign}\left(w_{d+1} + \sum_{j=1}^d w_j x_j\right)$$

Assumption: data generated i.i.d. by some underlying distribution $P_{X,Y}$

Expected error: $\min_{w \in \mathbb{R}^{d+1}} \mathbb{E}(1_{Y([X^T \ 1]w) < 0})$ impossible! $P_{X,Y}$ unknown

Empirical error (EE): $\min_w \frac{1}{N} \sum_{i=1}^N h(\underbrace{y_i ([x^T \ 1]w)}_{\text{margin}})$, where $h(z) = 1_{z < 0}$.

Convexification: EE neither convex nor differentiable (NP-hard problem).

Solution: replace $h : \mathbb{R} \rightarrow \{0, 1\}$ with convex loss $L : \mathbb{R} \rightarrow \mathbb{R}_+$.

Criterion: $\min_w \underbrace{\sum_{i=1}^N L(\underbrace{y_i (w^T x_i + b)}_{\text{margin}})}_{f(w)} + \tau \psi(w)$

Regularizer: $\psi = \ell_1 \Rightarrow$ encourage sparseness \Rightarrow feature selection

Convex losses: $L : \mathbb{R} \rightarrow \mathbb{R}_+$ is a (preferably convex) loss function.

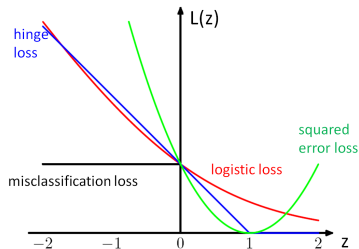
Machine/Statistical Learning: Linear Classification

Criterion:
$$\min_w \underbrace{\sum_{i=1}^N L(\underbrace{y_i (w^T x_i + b)}_{\text{margin}})}_{f(w)} + \tau \psi(w)$$

Regularizer: $\psi = \ell_1 \Rightarrow$ encourage sparseness \Rightarrow feature selection

Convex losses: $L : \mathbb{R} \rightarrow \mathbb{R}_+$ is a (preferably convex) loss function.

- Misclassification loss: $L(z) = 1_{z < 0}$
- Hinge loss: $L(z) = \max\{1 - z, 0\}$
- Logistic loss: $L(z) = \frac{\log(1 + \exp(-z))}{\log 2}$
- Squared loss: $L(z) = (z - 1)^2$



Machine/Statistical Learning: General Formulation

This formulation cover a wide range of **linear** ML methods:

$$\min_w \underbrace{\sum_{i=1}^N L(y_i ([x^T \ 1]w))}_{f(w)} + \tau \psi(w)$$

- Least squares regression: $L(z) = (z - 1)^2$, $\psi(w) = 0$.
- Ridge regression: $L(z) = (z - 1)^2$, $\psi(w) = \|w\|_2^2$.
- Lasso regression: $L(z) = (z - 1)^2$, $\psi(w) = \|w\|_1$
- Logistic regression: $L(z) = \log(1 + \exp(-z))$ (ridge, if $\psi(w) = \|w\|_2^2$)
- Sparse logistic regression: $L(z) = \log(1 + \exp(-z))$, $\psi(w) = \|w\|_1$
- Support vector machines: $L(z) = \max\{1 - z, 0\}$, $\psi(w) = \|w\|_2^2$
- Boosting: $L(z) = \exp(-z)$,
- ...

Machine/Statistical Learning: Nonlinear Problems

What about **non-linear** functions?

Simply use $\hat{y} = \phi(x, w) = \sum_{j=1}^D w_j \phi_j(x)$, where $\phi_j : \mathbb{R}^d \rightarrow \mathbb{R}$

Essentially, nothing changes; **computationally, a lot may change!**

$$\min_w \underbrace{\sum_{i=1}^N L(y_i \phi(x, w))}_{f(w)} + \tau \psi(w)$$

Key feature: $\phi(x, w)$ is **still linear** with respect to w , thus f inherits the **convexity** of L .

Examples: polynomials, radial basis functions, wavelets, splines, kernels,...

Recover the linear case, letting $D = d + 1$, $f_j(x) = x_j$, and $f_{d+1} = 1$.

Structured Sparsity

ℓ_1 regularization promotes **sparsity**

A very simple sparsity pattern: prefer models with **small cardinality**

Structured Sparsity

ℓ_1 regularization promotes **sparsity**

A very simple sparsity pattern: prefer models with **small cardinality**

Can we promote less trivial sparsity patterns? How?

Structured Sparsity

ℓ_1 regularization promotes **sparsity**

A very simple sparsity pattern: prefer models with **small cardinality**

Can we promote less trivial sparsity patterns? How?



Structured Sparsity

ℓ_1 regularization promotes **sparsity**

A very simple sparsity pattern: prefer models with **small cardinality**

Can we promote less trivial sparsity patterns? How?



Group/structured regularization.

Structured Sparsity and Groups

Main goal: to promote **structural patterns**, not just penalize cardinality

Structured Sparsity and Groups

Main goal: to promote **structural patterns**, not just penalize cardinality

Group sparsity: discard/keep entire *groups* of features (Bach et al., 2012)

- **density** inside each group
- **sparsity** with respect to the groups which are selected
- choice of groups: prior knowledge about the intended *sparsity patterns*

Structured Sparsity and Groups

Main goal: to promote **structural patterns**, not just penalize cardinality

Group sparsity: discard/keep entire *groups* of features (Bach et al., 2012)

- **density** inside each group
- **sparsity** with respect to the groups which are selected
- choice of groups: prior knowledge about the intended *sparsity patterns*

Yields statistical gains if the assumption is correct (Stojnic et al., 2009)

Structured Sparsity and Groups

Main goal: to promote **structural patterns**, not just penalize cardinality

Group sparsity: discard/keep entire *groups* of features (Bach et al., 2012)

- **density** inside each group
- **sparsity** with respect to the groups which are selected
- choice of groups: prior knowledge about the intended *sparsity patterns*

Yields statistical gains if the assumption is correct (Stojnic et al., 2009)

Many applications:

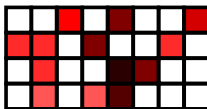
- feature template selection (Martins et al., 2011)
- multi-task learning (Caruana, 1997; Obozinski et al., 2010)
- learning the structure of graphical models (Schmidt and Murphy, 2010)

“Grid” Sparsity

For feature spaces that can be arranged as a grid (examples next)



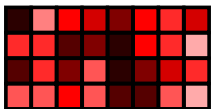
dense



sparse

“Grid” Sparsity

For feature spaces that can be arranged as a grid (examples next)



dense



sparse



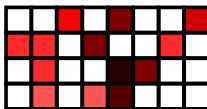
group sparse

“Grid” Sparsity

For feature spaces that can be arranged as a grid (examples next)



dense



sparse



group sparse

Goal: push *entire columns* to have zero weights

The groups are the columns of the grid

Example: Sparsity with Multiple Classes

In multi-class (more than just 2 classes) classification, a common formulation is

$$\hat{y} = \arg \max_{y \in \{1, \dots, K\}} x^T w_y$$

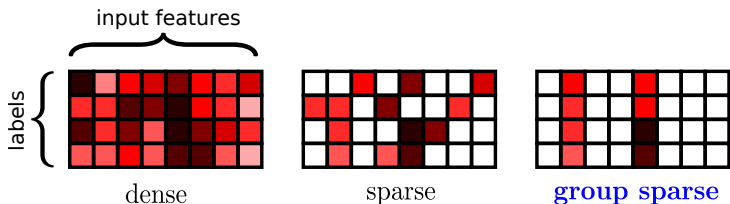
Weight vector $w = (w_1, \dots, w_K) \in \mathbb{R}^{Kd}$ has a natural group/grid organization:

Example: Sparsity with Multiple Classes

In multi-class (more than just 2 classes) classification, a common formulation is

$$\hat{y} = \arg \max_{y \in \{1, \dots, K\}} x^T w_y$$

Weight vector $w = (w_1, \dots, w_K) \in \mathbb{R}^{Kd}$ has a natural group/grid organization:



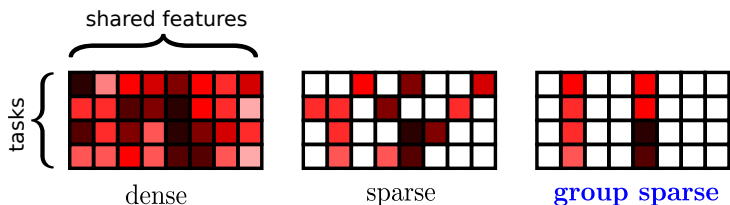
Simple sparsity is wasteful: may still need to keep all the features

Structured sparsity: discard some input features (feature selection)

Example: Multi-Task Learning

Same thing, except now rows are **tasks** and columns are **features**

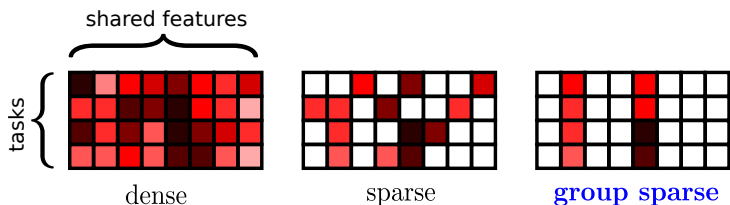
Example: simultaneous regression (seek function into $\mathbb{R}^d \rightarrow \mathbb{R}^b$)



Example: Multi-Task Learning

Same thing, except now rows are **tasks** and columns are **features**

Example: simultaneous regression (seek function into $\mathbb{R}^d \rightarrow \mathbb{R}^b$)

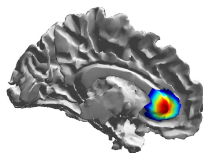


Goal: discard features that are irrelevant for *all* tasks

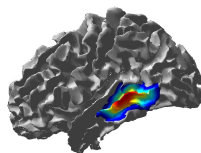
Approach: one group per feature (Caruana, 1997; Obozinski et al., 2010)

Example: Magnetoencephalography (MEG)

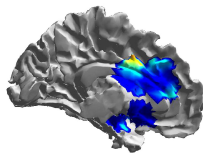
Group: localized cortex area at localized time period (Bolstad et al., 2009)



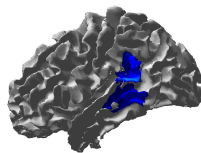
(a)



(b)

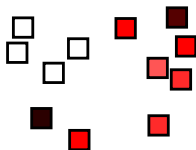


(c)



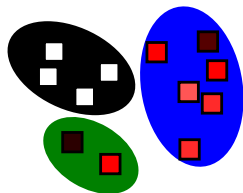
(d)

Group Sparsity



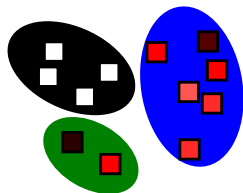
- D features

Group Sparsity



- D features
- M groups G_1, \dots, G_M , each $G_m \subseteq \{1, \dots, D\}$
- parameter subvectors x_1, \dots, x_M

Group Sparsity

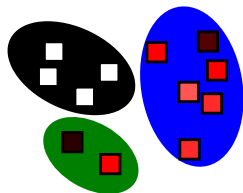


- D features
- M groups G_1, \dots, G_M , each $G_m \subseteq \{1, \dots, D\}$
- parameter subvectors x_1, \dots, x_M

Group-Lasso (Bakin, 1999; Yuan and Lin, 2006):

$$\psi(x) = \sum_{m=1}^M \|x_{G_m}\|_2$$

Group Sparsity



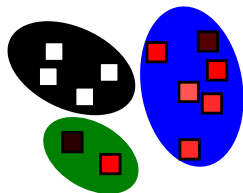
- D features
- M groups G_1, \dots, G_M , each $G_m \subseteq \{1, \dots, D\}$
- parameter subvectors x_1, \dots, x_M

Group-Lasso (Bakin, 1999; Yuan and Lin, 2006):

$$\psi(x) = \sum_{m=1}^M \|x_{G_m}\|_2$$

- Intuitively: the ℓ_1 **norm** of the ℓ_2 **norms**
- Technically, still a norm (called a *mixed* norm, denoted $\ell_{2,1}$)

Group Sparsity



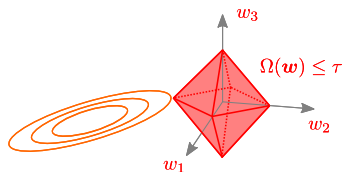
- D features
- M groups G_1, \dots, G_M , each $G_m \subseteq \{1, \dots, D\}$
- parameter subvectors x_1, \dots, x_M

Group-Lasso (Bakin, 1999; Yuan and Lin, 2006):

$$\psi(x) = \sum_{m=1}^M \lambda_m \|x_{G_m}\|_2$$

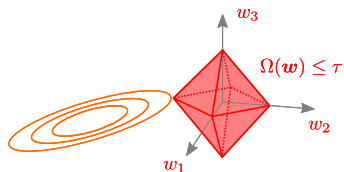
- Intuitively: the ℓ_1 **norm** of the ℓ_2 **norms**
- Technically, still a norm (called a *mixed* norm, denoted $\ell_{2,1}$)
- Weighted version: λ_m are prior weights for groups (groups may have different sizes)

Lasso versus group-Lasso

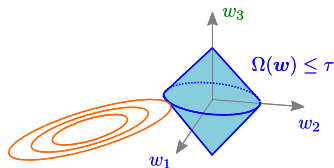


$$\Omega(\mathbf{w}) = |w_1| + |w_2| + |w_3|$$

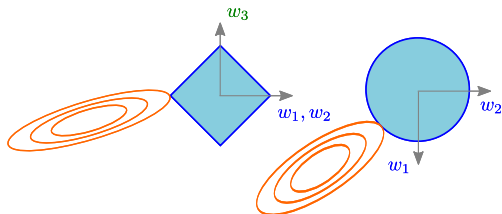
Lasso versus group-Lasso



$$\Omega(\mathbf{w}) = |w_1| + |w_2| + |w_3|$$



$$\Omega(\mathbf{w}) = \sqrt{w_1^2 + w_2^2} + |w_3|$$



Composite Absolute Penalties (Zhao et al., 2009)

A mixed-norm regularization:

$$\psi(\mathbf{x}) = \left(\sum_{m=1}^M \|\mathbf{x}_m\|_q^r \right)^{1/r}$$

The r -norm of the q -norms ($r \geq 1, q \geq 1$)

Technically, this is also a norm, called a **mixed norm**, denoted $\ell_{q,r}$

Composite Absolute Penalties (Zhao et al., 2009)

A mixed-norm regularization:

$$\psi(\mathbf{x}) = \left(\sum_{m=1}^M \|\mathbf{x}_m\|_q^r \right)^{1/r}$$

The r -norm of the q -norms ($r \geq 1, q \geq 1$)

Technically, this is also a norm, called a **mixed norm**, denoted $\ell_{q,r}$

- The most common choice: $\ell_{2,1}$ norm
- Another frequent choice: $\ell_{\infty,1}$ norm (Quattoni et al., 2009; Graça et al., 2009; Eisenstein et al., 2011; Wright et al., 2009)

Three Scenarios

- Non-overlapping Groups
- Tree-structured Groups
- Graph-structured Groups

Non-overlapping Groups

Assume that G_1, \dots, G_M (where $G_m \subset \{1, \dots, d\}$) constitute a partition:

$$\bigcup_{i=1}^M G_m = \{1, \dots, d\} \quad \text{and} \quad i \neq j \Rightarrow G_i \cap G_j = \emptyset$$

Non-overlapping Groups

Assume that G_1, \dots, G_M (where $G_m \subset \{1, \dots, d\}$) constitute a partition:

$$\bigcup_{i=1}^M G_m = \{1, \dots, d\} \quad \text{and} \quad i \neq j \Rightarrow G_i \cap G_j = \emptyset$$

$$\psi(x) = \sum_{m=1}^M \lambda_m \|x_{G_m}\|_2$$

Trivial choices of groups recover *unstructured* regularizers:

Non-overlapping Groups

Assume that G_1, \dots, G_M (where $G_m \subset \{1, \dots, d\}$) constitute a partition:

$$\bigcup_{i=1}^M G_m = \{1, \dots, d\} \quad \text{and} \quad i \neq j \Rightarrow G_i \cap G_j = \emptyset$$

$$\psi(x) = \sum_{m=1}^M \lambda_m \|x_{G_m}\|_2$$

Trivial choices of groups recover *unstructured* regularizers:

- ℓ_2 -regularization: one large group $G_1 = \{1, \dots, d\}$
- ℓ_1 -regularization: d singleton groups $G_m = \{m\}$

Non-overlapping Groups

Assume that G_1, \dots, G_M (where $G_m \subset \{1, \dots, d\}$) constitute a partition:

$$\bigcup_{i=1}^M G_m = \{1, \dots, d\} \quad \text{and} \quad i \neq j \Rightarrow G_i \cap G_j = \emptyset$$

$$\psi(x) = \sum_{m=1}^M \lambda_m \|x_{G_m}\|_2$$

Trivial choices of groups recover *unstructured* regularizers:

- ℓ_2 -regularization: one large group $G_1 = \{1, \dots, d\}$
- ℓ_1 -regularization: d singleton groups $G_m = \{m\}$

Examples of non-trivial groups:

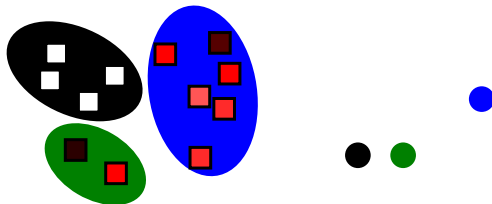
- label-based groups
- task-based groups

Tree-Structured Groups

Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)

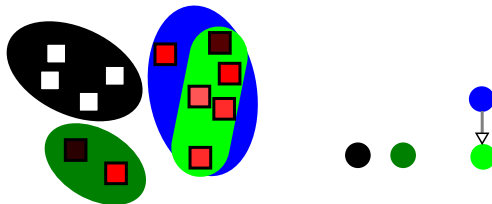
Tree-Structured Groups

Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)



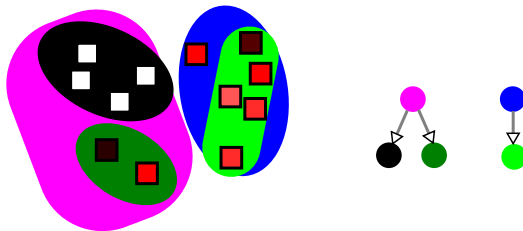
Tree-Structured Groups

Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)



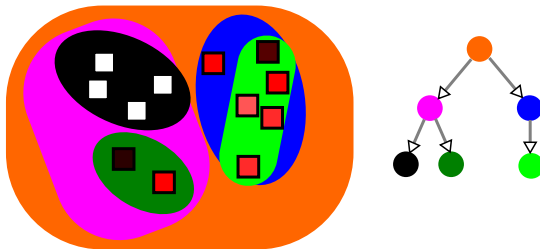
Tree-Structured Groups

Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)



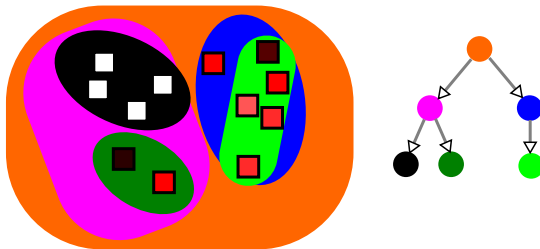
Tree-Structured Groups

Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)



Tree-Structured Groups

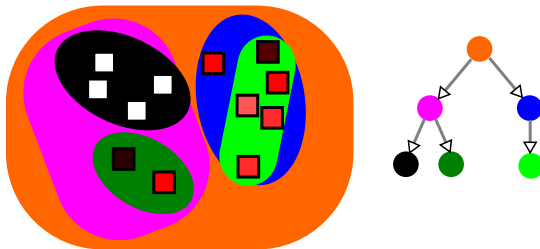
Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)



- What is the **sparsity pattern**?

Tree-Structured Groups

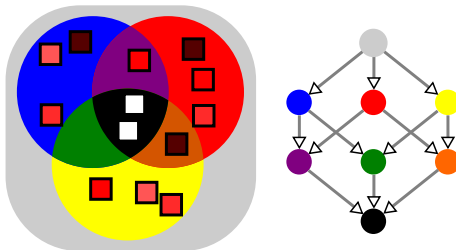
Assumption: if two groups overlap, one is contained in the other
⇒ **hierarchical** structure (Kim and Xing, 2010; Mairal et al., 2010)



- What is the **sparsity pattern**?
- If a group is discarded, all its descendants are also discarded

Graph-Structured Groups

In general: groups can be represented as a **directed acyclic graph**



- set inclusion induces a **partial order** on groups (Jenatton et al., 2009)
- feature space becomes a **poset**
- **sparsity patterns**: given by this poset

Sparsest solution:

- From $Bx = b \in \mathbb{R}^p$, find $x \in \mathbb{R}^n$ ($p < n$).
- $\min_x \|x\|_0$ s.t. $Bx = b$
- Yields exact solution, under some conditions.

Matrix Inference Problems

Sparsest solution:

- From $Bx = b \in \mathbb{R}^p$, find $x \in \mathbb{R}^n$ ($p < n$).
- $\min_x \|x\|_0$ s.t. $Bx = b$
- Yields exact solution, under some conditions.

Lowest rank solution:

- From $\mathcal{B}(X) = b \in \mathbb{R}^p$, find $X \in \mathbb{R}^{m \times n}$ ($p < mn$).
- $\min_X \text{rank}(X)$ s.t. $\mathcal{B}(X) = b$
- Yields exact solution, under some conditions.

Both *NP*–hard (in general); the same is true of noisy versions:

$$\min_{X \in \mathbb{R}^{m \times n}} \text{rank}(X) \text{ s.t. } \|\mathcal{B}(X) - b\|_2^2$$

Matrix Inference Problems

Sparsest solution:

- From $Bx = b \in \mathbb{R}^p$, find $x \in \mathbb{R}^n$ ($p < n$).
- $\min_x \|x\|_0$ s.t. $Bx = b$
- Yields exact solution, under some conditions.

Lowest rank solution:

- From $\mathcal{B}(X) = b \in \mathbb{R}^p$, find $X \in \mathbb{R}^{m \times n}$ ($p < mn$).
- $\min_X \text{rank}(X)$ s.t. $\mathcal{B}(X) = b$
- Yields exact solution, under some conditions.

Both *NP*–hard (in general); the same is true of noisy versions:

$$\min_{X \in \mathbb{R}^{m \times n}} \text{rank}(X) \text{ s.t. } \|\mathcal{B}(X) - b\|_2^2$$

Under some conditions, the **same solution** is obtained by replacing $\text{rank}(X)$ by the **nuclear norm** $\|X\|_*$ (as any norm, it is convex) (Recht et al., 2010)

Matrix Nuclear Norm (and Other Norms)

- Also known as **trace norm**; the **ℓ_1 -type norm** for matrices $X \in \mathbb{R}^{m \times n}$

- Definition: $\|X\|_* = \text{trace}(\sqrt{X^T X}) = \sum_{i=1}^{\min\{m,n\}} \sigma_i$,
the σ_i are the **singular values** of X .

Matrix Nuclear Norm (and Other Norms)

- Also known as **trace norm**; the **ℓ_1 -type norm** for matrices $X \in \mathbb{R}^{m \times n}$

- Definition: $\|X\|_* = \text{trace}(\sqrt{X^T X}) = \sum_{i=1}^{\min\{m,n\}} \sigma_i$,
the σ_i are the **singular values** of X .

- Particular case of **Schatten** q -norm: $\|X\|_q = \left(\sum_{i=1}^{\min\{m,n\}} (\sigma_i)^q \right)^{1/q}$.

Matrix Nuclear Norm (and Other Norms)

- Also known as **trace norm**; the **ℓ_1 -type norm** for matrices $X \in \mathbb{R}^{m \times n}$

- Definition: $\|X\|_* = \text{trace}(\sqrt{X^T X}) = \sum_{i=1}^{\min\{m,n\}} \sigma_i$,
the σ_i are the **singular values** of X .

- Particular case of **Schatten** q -norm: $\|X\|_q = \left(\sum_{i=1}^{\min\{m,n\}} (\sigma_i)^q \right)^{1/q}$.
- Two other notable Schatten norms:

- Frobenius norm**: $\|X\|_2 = \|X\|_F = \sqrt{\sum_{i=1}^{\min\{m,n\}} (\sigma_i)^2} = \sqrt{\sum_{i,j} X_{i,j}^2}$
- Spectral norm**: $\|X\|_\infty = \max\{\sigma_1, \dots, \sigma_{\min\{m,n\}}\}$

Nuclear Norm Regularization

Tikhonov formulation: $\min_X \underbrace{\|\mathcal{B}(X) - b\|_2^2}_{f(X)} + \underbrace{\tau \|X\|_*}_{\tau\psi(X)}$

Nuclear Norm Regularization

Tikhonov formulation: $\min_X \underbrace{\|\mathcal{B}(X) - b\|_2^2}_{f(X)} + \underbrace{\tau \|X\|_*}_{\tau\psi(X)}$

Linear observations: $\mathcal{B} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$, $(\mathcal{B}(X))_i = \langle B_{(i)}, X \rangle$,

$$B_{(i)} \in \mathbb{R}^{m \times n}, \text{ and } \langle B, X \rangle = \sum_{ij} B_{ij} X_{ij} = \text{trace}(B^T X)$$

Nuclear Norm Regularization

Tikhonov formulation: $\min_X \underbrace{\|\mathcal{B}(X) - b\|_2^2}_{f(X)} + \underbrace{\tau \|X\|_*}_{\tau\psi(X)}$

Linear observations: $\mathcal{B} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$, $(\mathcal{B}(X))_i = \langle B_{(i)}, X \rangle$,

$$B_{(i)} \in \mathbb{R}^{m \times n}, \text{ and } \langle B, X \rangle = \sum_{ij} B_{ij} X_{ij} = \text{trace}(B^T X)$$

Particular case: **matrix completion**, each matrix $B_{(i)}$ has one 1 and is zero everywhere else.

Nuclear Norm Regularization

Tikhonov formulation: $\min_X \underbrace{\|\mathcal{B}(X) - b\|_2^2}_{f(X)} + \underbrace{\tau \|X\|_*}_{\tau\psi(X)}$

Linear observations: $\mathcal{B} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$, $(\mathcal{B}(X))_i = \langle B_{(i)}, X \rangle$,

$$B_{(i)} \in \mathbb{R}^{m \times n}, \text{ and } \langle B, X \rangle = \sum_{ij} B_{ij} X_{ij} = \text{trace}(B^T X)$$

Particular case: **matrix completion**, each matrix $B_{(i)}$ has one 1 and is zero everywhere else.

Why does the **nuclear norm** favor **low rank** solutions?

Nuclear Norm Regularization

Tikhonov formulation: $\min_X \underbrace{\|\mathcal{B}(X) - b\|_2^2}_{f(X)} + \underbrace{\tau \|X\|_*}_{\tau \psi(X)}$

Linear observations: $\mathcal{B} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$, $(\mathcal{B}(X))_i = \langle B_{(i)}, X \rangle$,

$$B_{(i)} \in \mathbb{R}^{m \times n}, \text{ and } \langle B, X \rangle = \sum_{ij} B_{ij} X_{ij} = \text{trace}(B^T X)$$

Particular case: **matrix completion**, each matrix $B_{(i)}$ has one 1 and is zero everywhere else.

Why does the **nuclear norm** favor **low rank** solutions? Let $Y = U\Lambda V^T$ be the singular value decomposition, where $\Lambda = \text{diag}(\sigma_1, \dots, \sigma_{\min\{m,n\}})$; then

$$\arg \min_X \frac{1}{2} \|Y - X\|_F^2 + \tau \|X\|_* = U \underbrace{\text{soft}(X, \tau)}_{\text{may yield zeros}} V^T$$

...**singular value thresholding** (Ma et al., 2011; Cai et al., 2010)

Another Matrix Inference Problem: Inverse Covariance

Consider n samples $y_1, \dots, y_n \in \mathbb{R}^d$ of a **Gaussian** r.v. $Y \sim \mathcal{N}(\mu, C)$; the log-likelihood is

$$L(P) = \log p(y_1, \dots, y_n | P) = \log \det(P) - \text{trace}(SP) + \text{constant}$$

where $S = \frac{1}{n} \sum_{i=1}^n (y_i - \mu)(y_i - \mu)^T$ and $P = C^{-1}$ (**inverse covariance**).

Another Matrix Inference Problem: Inverse Covariance

Consider n samples $y_1, \dots, y_n \in \mathbb{R}^d$ of a **Gaussian** r.v. $Y \sim \mathcal{N}(\mu, C)$; the log-likelihood is

$$L(P) = \log p(y_1, \dots, y_n | P) = \log \det(P) - \text{trace}(SP) + \text{constant}$$

where $S = \frac{1}{n} \sum_{i=1}^n (y_i - \mu)(y_i - \mu)^T$ and $P = C^{-1}$ (**inverse covariance**).

Zeros in P reveal **conditional independencies** between components of Y :

$$P_{ij} = 0 \Leftrightarrow Y_i \perp\!\!\!\perp Y_j | \{Y_k, k \neq i, j\}$$

...exploited to infer (in)dependencies among Gaussian variables. Widely used in computational biology and neuroscience, social network analysis, ...

Another Matrix Inference Problem: Inverse Covariance

Consider n samples $y_1, \dots, y_n \in \mathbb{R}^d$ of a **Gaussian** r.v. $Y \sim \mathcal{N}(\mu, C)$; the log-likelihood is

$$L(P) = \log p(y_1, \dots, y_n | P) = \log \det(P) - \text{trace}(SP) + \text{constant}$$

where $S = \frac{1}{n} \sum_{i=1}^n (y_i - \mu)(y_i - \mu)^T$ and $P = C^{-1}$ (**inverse covariance**).

Zeros in P reveal **conditional independencies** between components of Y :

$$P_{ij} = 0 \Leftrightarrow Y_i \perp\!\!\!\perp Y_j | \{Y_k, k \neq i, j\}$$

...exploited to infer (in)dependencies among Gaussian variables. Widely used in computational biology and neuroscience, social network analysis, ...

Sparsity (presence of zeros) in P is encouraged by solving

$$\min_{P \succ 0} \underbrace{-\log \det(P) + \text{trace}(SP)}_{f(P)} + \tau \underbrace{\|\text{vect}(P)\|_1}_{\psi(P)}$$

where $\text{vect}(P) = [P_{1,1}, \dots, P_{d,d}]^T$.

Atomic-Norm Regularization

Key concept in sparse modeling: synthesize “object” using a few **atoms**:

$$x = \sum_{i=1}^{|\mathcal{A}|} c_i a_i$$

- \mathcal{A} is the set of **atoms** (the **atomic set**), or building blocks.
- $c_i \geq 0$ are weights; x is **simple/sparse** object $\Rightarrow \|c\|_0 \ll |\mathcal{A}|$
- Formally, \mathcal{A} is a compact subset of \mathbb{R}^n

Atomic-Norm Regularization

Key concept in sparse modeling: synthesize “object” using a few **atoms**:

$$x = \sum_{i=1}^{|\mathcal{A}|} c_i a_i$$

- \mathcal{A} is the set of **atoms** (the **atomic set**), or building blocks.
- $c_i \geq 0$ are weights; x is **simple/sparse** object $\Rightarrow \|c\|_0 \ll |\mathcal{A}|$
- Formally, \mathcal{A} is a compact subset of \mathbb{R}^n

The (Minkowski) **gauge** of \mathcal{A} is:

$$\|x\|_{\mathcal{A}} = \inf \{ t > 0 : x \in t \operatorname{conv}(\mathcal{A}) \}$$

Assuming that \mathcal{A} centrally symmetry about the origin ($a \in \mathcal{A} \Rightarrow -a \in \mathcal{A}$), $\|\cdot\|_{\mathcal{A}}$ is a norm, called the **atomic norm** Chandrasekaran et al. (2012).

Atomic-Norm Regularization

The atomic norm

$$\begin{aligned}\|x\|_{\mathcal{A}} &= \inf \{ t > 0 : x \in t \operatorname{conv}(\mathcal{A}) \} \\ &= \inf \left\{ \sum_{i=1}^{|\mathcal{A}|} c_i : x = \sum_{i=1}^{|\mathcal{A}|} c_i a_i, c_i \geq 0 \right\}\end{aligned}$$

...assuming that the centroid of \mathcal{A} is at the origin.

Atomic-Norm Regularization

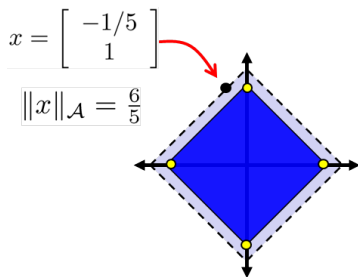
The atomic norm

$$\begin{aligned}\|x\|_{\mathcal{A}} &= \inf \{t > 0 : x \in t \operatorname{conv}(\mathcal{A})\} \\ &= \inf \left\{ \sum_{i=1}^{|\mathcal{A}|} c_i : x = \sum_{i=1}^{|\mathcal{A}|} c_i a_i, c_i \geq 0 \right\}\end{aligned}$$

...assuming that the centroid of \mathcal{A} is at the origin.

Example: the ℓ_1 norm as an atomic norm

- $\mathcal{A} = \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \end{bmatrix} \right\}$
- $\operatorname{conv}(\mathcal{A}) = B_1(1)$ (ℓ_1 unit ball).
- $\|x\|_{\mathcal{A}} = \inf \{t > 0 : x \in t B_1(1)\}$
 $= \|x\|_1$



Atomic Norms: More Examples

Examples with easy forms:

- sparse vectors*

$$\mathcal{A} = \{\pm e_i\}_{i=1}^N$$

$$\text{conv}(\mathcal{A}) = \text{cross-polytope}$$

$$\|x\|_{\mathcal{A}} = \|x\|_1$$

- low-rank matrices*

$$\mathcal{A} = \{A : \text{rank}(A) = 1, \|A\|_F = 1\}$$

$$\text{conv}(\mathcal{A}) = \text{nuclear norm ball}$$

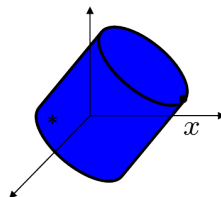
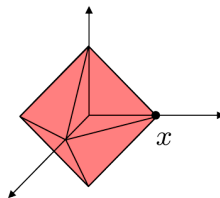
$$\|x\|_{\mathcal{A}} = \|x\|_{\star}$$

- binary vectors*

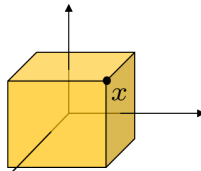
$$\mathcal{A} = \{\pm 1\}^N$$

$$\text{conv}(\mathcal{A}) = \text{hypercube}$$

$$\|x\|_{\mathcal{A}} = \|x\|_{\infty}$$



*symmetric
matrices



Atomic-Norm Regularization

Given an **atomic set** \mathcal{A} , we can adopt an Ivanov formulation

$$\min f(x) \quad \text{s.t.} \quad \|x\|_{\mathcal{A}} \leq \delta$$

(for some $\delta > 0$) tends to recover x with sparse atomic representation.

Atomic-Norm Regularization

Given an **atomic set** \mathcal{A} , we can adopt an Ivanov formulation

$$\min f(x) \quad \text{s.t.} \quad \|x\|_{\mathcal{A}} \leq \delta$$

(for some $\delta > 0$) tends to recover x with sparse atomic representation.

Can formulate algorithms for the various special cases — but is a **general approach** available for this formulation?

Atomic-Norm Regularization

Given an **atomic set** \mathcal{A} , we can adopt an Ivanov formulation

$$\min f(x) \quad \text{s.t.} \quad \|x\|_{\mathcal{A}} \leq \delta$$

(for some $\delta > 0$) tends to recover x with sparse atomic representation.

Can formulate algorithms for the various special cases — but is a **general approach** available for this formulation?

Yes! The **conditional gradient!** (more later.)

It is also possible to tackle the **Tikhonov and Morozov** formulations

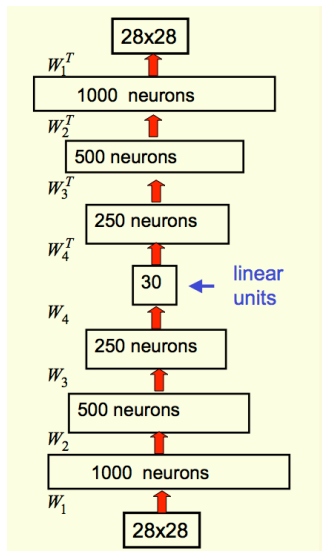
$$\min_x f(x) + \tau \|x\|_{\mathcal{A}} \quad \text{and} \quad \min_x \|x\|_{\mathcal{A}} \quad \text{s.t.} \quad f(x) \leq \epsilon$$

(more later).

There is much recent interest in **deep learning** (a.k.a. **deep belief networks** (DBN) or **deep neural networks**) — very popular in speech and image processing.

- Based heavily on neural networks (70s and 80s). Idea is to mimic the neuron interconnections in a cortex.
- DBN transforms the data vector into another data vector, via a highly structured (usually layered) sequence of simple operations.
- The operations at each layer are simple (e.g. linear transformation, logistic function) but their composition is nonconvex.
- **Aim:** Make the transformed data vector (the output from the DBN) easier to use in learning tasks than the original data vector. e.g. can apply SVN on the transformed vector, or thresholding of a single output, or max of multiple outputs.
- Use labelled data vectors to train the network i.e. choose the parameters of the transformations at each layer.

Deep Learning: Example



Example of a deep belief network for autoencoding (Hinton, 2007). Output (at top) depends on input (at bottom) of an image with 28×28 pixels. Transformations parametrized by W_1, W_2, W_3, W_4 ; output is a highly nonlinear function of these parameters.

Deep learning problems have separable, nonconvex objectives.

- There is one loss term f_i for each labelled data vector.
- Unknowns are the parameters of the network.
- Commonly use heuristics such as **pretraining**: fix the parameters in **all but one** layer, and solve just for these parameters.
- Regularization terms are used to induce sparsity or structure on the parameters.
- **Stochastic gradient** algorithms are a workhorse approach. **Quasi-Newton** and **inexact Newton** methods based on batches of data have also been applied with success.

- Many inference, learning, signal/image processing problems can be formulated as optimization problems.
- Sparsity-inducing regularizers play an important role in these problems
- There are several way to induce sparsity
- It is possible to formulate structured sparsity
- It is possible to extend the sparsity rationale to other objects, namely matrices
- Atomic norms provide a unified framework for sparsity/simplicity regularization

References I

- Amaldi, E. and Kann, V. (1998). On the approximation of minimizing non zero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209:237–260.
- Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2012). Structured sparsity through convex optimization. *Statistical Science*, 27:450–468.
- Bakin, S. (1999). *Adaptive regression and model selection in data mining problems*. PhD thesis, Australian National University.
- Bolstad, A., Veen, B. V., and Nowak, R. (2009). Space-time event sparse penalization for magnetoencephalography. *NeuroImage*, 46:1066–1081.
- Cai, J.-F., Candès, E., and Shen, Z. (2010). A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20:1956–1982.
- Candès, E., Romberg, J., and Tao, T. (2006a). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52:489–509.
- Candès, E., Romberg, J., and Tao, T. (2006b). Stable signal recovery from incomplete and inaccurate measurements. *Communications in Pure and Applied Mathematics*, 59:1207–1223.
- Caruana, R. (1997). Multitask learning. *Machine Learning*, 28(1):41–75.
- Chandrasekaran, V., Recht, B., Parrilo, P., and Willsky, A. (2012). The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12:805–849.

References II

- Chen, S., Donoho, D., and Saunders, M. (1995). Atomic decomposition by basis pursuit. Technical report, Department of Statistics, Stanford University.
- Davis, G., Mallat, S., and Avellaneda, M. (1997). Greedy adaptive approximation. *Journal of Constructive Approximation*, 13:57–98.
- Donoho, D. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52:1289–1306.
- Eisenstein, J., Smith, N. A., and Xing, E. P. (2011). Discovering sociolinguistic associations with structured sparsity. In *Proc. of ACL*.
- Garnaev, A. and Gluskin, E. (1984). The widths of an Euclidean ball. *Doklady Akademii Nauk*, 277:1048–1052.
- Graça, J., Ganchev, K., Taskar, B., and Pereira, F. (2009). Posterior vs. parameter sparsity in latent variable models. *Advances in Neural Information Processing Systems*.
- Haupt, J. and Nowak, R. (2006). Signal reconstruction from noisy random projections. *IEEE Transactions on Information Theory*, 52:4036–4048.
- Jenatton, R., Audibert, J.-Y., and Bach, F. (2009). Structured variable selection with sparsity-inducing norms. Technical report, arXiv:0904.3523.
- Kashin, B. (1977). Diameters of certain finite-dimensional sets in classes of smooth functions. *Izvestiya Akademii Nauk. SSSR: Seriya Matematicheskaya*, 41:334–351.
- Kim, S. and Xing, E. (2010). Tree-guided group lasso for multi-task regression with structured sparsity. In *Proc. of ICML*.

References III

- Ma, S., Goldfarb, D., and Chen, L. (2011). Fixed point and Bregman iterative methods for matrix rank minimization. *Mathematical Programming (Series A)*, 128:321–353.
- Mairal, J., Jenatton, R., Obozinski, G., and Bach, F. (2010). Network flow algorithms for structured sparsity. In *Advances in Neural Information Processing Systems*.
- Martins, A. F. T., Smith, N. A., Aguiar, P. M. Q., and Figueiredo, M. A. T. (2011). Structured Sparsity in Structured Prediction. In *Proc. of Empirical Methods for Natural Language Processing*.
- Muthukrishnan, S. (2005). *Data Streams: Algorithms and Applications*. Now Publishers, Boston, MA.
- Obozinski, G., Taskar, B., and Jordan, M. (2010). Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252.
- Quattoni, A., Carreras, X., Collins, M., and Darrell, T. (2009). An efficient projection for $l_{1,\infty}$ regularization. In *Proc. of ICML*.
- Recht, B., Fazel, M., and Parrilo, P. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52:471–501.
- Schmidt, M. and Murphy, K. (2010). Convex structure learning in log-linear models: Beyond pairwise potentials. In *Proc. of AISTATS*.
- Stojnic, M., Parvaresh, F., and Hassibi, B. (2009). On the reconstruction of block-sparse signals with an optimal number of measurements. *Signal Processing, IEEE Transactions on*, 57(8):3075–3085.

- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B.*, pages 267–288.
- Tillmann, A. and Pfetsch, M. (2012). The computational complexity of RIP, NSP, and related concepts in compressed sensing. Technical report, arXiv/1205.2081.
- Wright, S., Nowak, R., and Figueiredo, M. (2009). Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57:2479–2493.
- Yin, W. and Zhang, Y. (2008). Extracting salient features from less data via ℓ_1 -minimization authors. *SIAG/OPT Views-and-News*, 19:11–19.
- Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society (B)*, 68(1):49.
- Zhao, P., Rocha, G., and Yu, B. (2009). Grouped and hierarchical model selection through composite absolute penalties. *Annals of Statistics*, 37(6A):3468–3497.