Optimization methods for large-scale machine learning and sparse estimation

Julien Mairal

Inria Grenoble

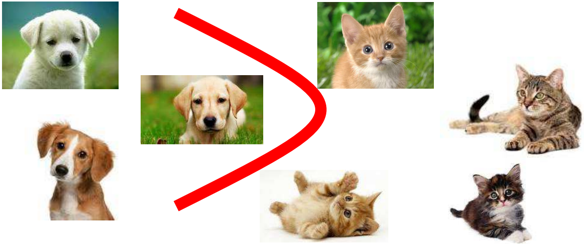
Nantes, Mascot-Num, 2018

Part II

Common paradigm: optimization for machine learning

Optimization is central to machine learning. For instance, in supervised learning, the goal is to learn a **prediction function** $f : \mathcal{X} \rightarrow \mathcal{Y}$ given labeled training data $(x_i, y_i)_{i=1,\ldots,n}$ with x_i in $\mathcal{X}$, and y_i in $\mathcal{Y}$:

$$\min_{f \in \mathcal{F}} \underbrace{\frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i))}_{\text{empirical risk, data fit}} + \underbrace{\lambda \Omega(f)}_{\text{regularization}} .$$



[Vapnik, 1995, Bottou, Curtis, and Nocedal, 2016]...

Paradigm 3: The sparsity principle

Let us consider again the classical scientific paradigm:

1. **observe** the world (gather data);
2. **propose models** of the world (design and learn);
3. **test** on new data (estimate the generalization error).

[Corfield et al., 2009].

Paradigm 3: The sparsity principle

Let us consider again the classical scientific paradigm:

1. **observe** the world (gather data);
2. **propose models** of the world (design and learn);
3. **test** on new data (estimate the generalization error).

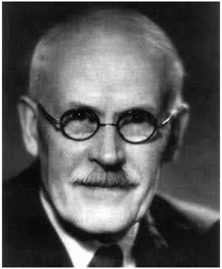
But...

- it is not always possible to distinguish the generalization error of various models based on available data.
- when a complex model A performs slightly better than a simple model B, should we prefer A or B?
- generalization error requires a predictive task: what about unsupervised learning? which measure should we use?
- we are also leaving aside the problem of non i.i.d. train/test data, biased data, testing with counterfactual reasoning...

[Corfield et al., 2009, Bottou et al., 2013, Schölkopf et al., 2012].

Paradigm 3: The sparsity principle

(a) Dorothy Wrinch
1894–1980



(b) Harold Jeffreys
1891–1989

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

[Wrinch and Jeffreys, 1921].

Paradigm 3: The sparsity principle

Remarks: sparsity is...

- appealing for experimental sciences for **model interpretation**;
- (too-)**well understood** in some mathematical contexts:

$$\min_{w \in \mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L\left(y_i, w^\top x_i\right)}_{\text{empirical risk, data fit}} + \underbrace{\lambda \|w\|_1}_{\text{regularization}}.$$

- extremely powerful for **unsupervised learning** in the context of matrix factorization, and **simple to use**.

[Olshausen and Field, 1996, Chen, Donoho, and Saunders, 1999, Tibshirani, 1996]...

Paradigm 3: The sparsity principle

Remarks: sparsity is...

- appealing for experimental sciences for **model interpretation**;
- (too-)**well understood** in some mathematical contexts:

$$\min_{w \in \mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L\left(y_i, w^\top x_i\right)}_{\text{empirical risk, data fit}} + \underbrace{\lambda \|w\|_1}_{\text{regularization}}.$$

- extremely powerful for **unsupervised learning** in the context of matrix factorization, and **simple to use**.

Today's challenges

- Develop sparse and **stable** (and **invariant?**) models.
- Go beyond clustering / low-rank / union of subspaces.

[Olshausen and Field, 1996, Chen, Donoho, and Saunders, 1999, Tibshirani, 1996]...

Some references

On kernel methods

- B. Schölkopf and A. J. Smola. Learning with kernels: support vector machines, regularization, optimization, and beyond. 2002.
- J. Shawe-Taylor and N. Cristianini. An introduction to support vector machines and other kernel-based learning methods. 2004.
- 635 slides:

http://members.cbio.mines-paristech.fr/~jvert/svn/kernelcourse/course/2018mva/

On sparse estimation

- M. Elad. Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. 2010.
- J. Mairal, F. Bach, and J. Ponce. Sparse Modeling for Image and Vision Processing. 2014. **free online**.

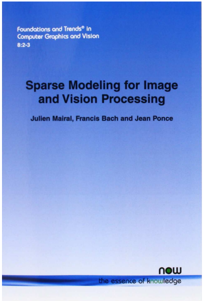
Some references

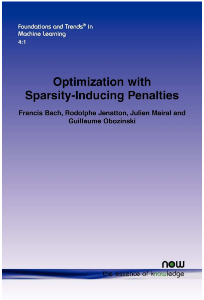
On large-scale optimization

- L. Bottou, F. E. Curtis and J. Nocedal. Optimization methods for large-scale machine learning, preprint arXiv:1606.04838, 2016.
- Y. Nesterov. Introductory lectures on convex optimization: A basic course. Springer .2013.
- S. Bubeck. Convex optimization: Algorithms and complexity. Foundations and Trends in Machine Learning. 2015.
- 387 slides by F. Bach: `http://www.di.ens.fr/~fbach/fbach_frejus_2017.pdf`.

Material on sparse estimation (freely available on arXiv)

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





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

Part I: Large-scale optimization for machine learning

Focus of this part

Minimizing large finite sums

Consider the minimization of a large sum of convex functions

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \psi(x) \right\},$$

where each f_i is L-**smooth and convex** and ψ is a convex regularization penalty but not necessarily differentiable.

Focus of this part

Minimizing large finite sums

Consider the minimization of a large sum of convex functions

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \psi(x) \right\},$$

where each f_i is **L-smooth and convex** and ψ is a convex regularization penalty but not necessarily differentiable.

Why this setting?

- convexity makes it easy to obtain **complexity** bounds.
- convex optimization is often effective for non-convex problems.

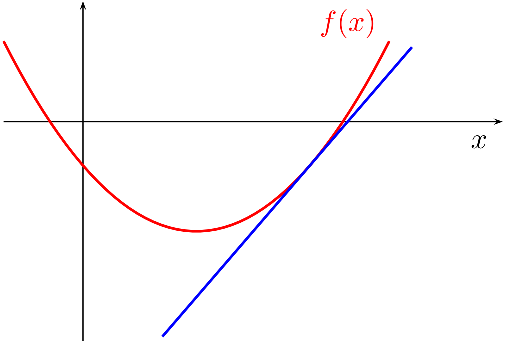
What we will not cover

- performance of approaches in terms of test error.

Introduction of a few optimization principles

Convex Functions

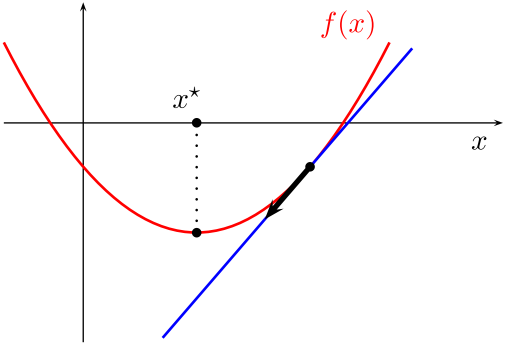
Why do we care about convexity?



Introduction of a few optimization principles

Convex Functions

Local observations give information about the global optimum

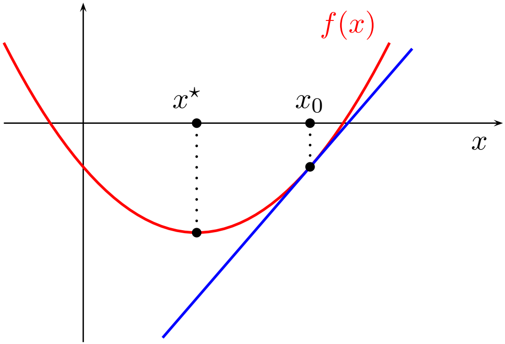


- $\nabla f(x) = 0$ is a necessary and sufficient optimality condition for differentiable convex functions;
- it is often easy to upper-bound $f(x) - f^\star$.

Introduction of a few optimization principles

An important inequality for L-smooth convex functions

If f is convex and smooth

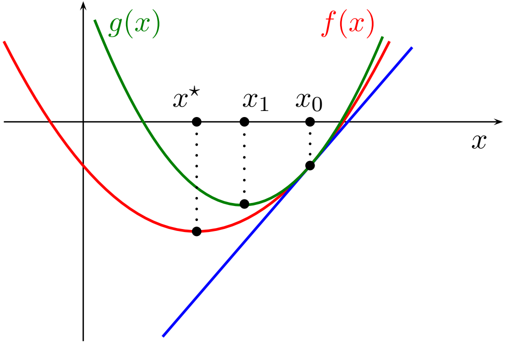


- $f(x) \geq \underbrace{f(x_0) + \nabla f(x_0)^\top (x - x_0)}_{\text{linear approximation}}$;
- if f is non-smooth, a similar inequality holds for subgradients.

Introduction of a few optimization principles

An important inequality for smooth functions

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

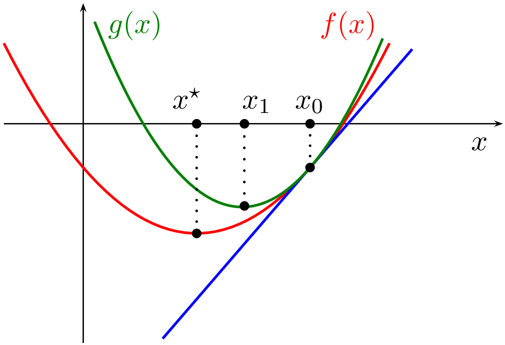


- $f(x) \leq g(x) = \underbrace{f(x_0) + \nabla f(x_0)^\top (x - x_0)}_{\text{linear approximation}} + \frac{L}{2}\|x - x_0\|_2^2$;

Introduction of a few optimization principles

An important inequality for smooth functions

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

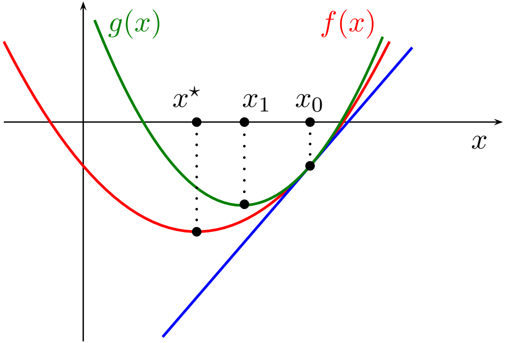


- $f(x) \leq g(x) = \underbrace{f(x_0) + \nabla f(x_0)^\top (x - x_0)}_{\text{linear approximation}} + \frac{L}{2}\|x - x_0\|_2^2$;
- $g(x) = C_{x_0} + \frac{L}{2}\|x_0 - (1/L)\nabla f(x_0) - x\|_2^2$.

Introduction of a few optimization principles

An important inequality for smooth functions

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



- $f(x) \leq g(x) = \underbrace{f(x_0) + \nabla f(x_0)^\top (x - x_0)}_{\text{linear approximation}} + \frac{L}{2}\|x - x_0\|_2^2$;
- $x_1 = x_0 - \frac{1}{L}\nabla f(x_0)$. (gradient descent step).

Introduction of a few optimization principles

Gradient Descent Algorithm

Assume that f is convex and L-smooth (∇f is L-Lipschitz).

Theorem

Consider the algorithm

$$\boxed{x_t \leftarrow x_{t-1} - \frac{1}{L}\nabla f(x_{t-1}).}$$

Then,

$$f(x_t) - f^\star \leq \frac{L\|x_0 - x^\star\|_2^2}{2t}.$$

Proof (1/2)

Proof of the main inequality for smooth functions

We want to show that for all x and z,

$$f(x) \leq f(z) + \nabla f(z)^\top (x - z) + \frac{L}{2}\|x - z\|_2^2.$$

Proof (1/2)

Proof of the main inequality for smooth functions

We want to show that for all x and z,

$$f(x) \leq f(z) + \nabla f(z)^\top (x - z) + \frac{L}{2}\|x - z\|_2^2.$$

By using Taylor's theorem with integral form,

$$f(x) - f(z) = \int_0^1 \nabla f(tx + (1-t)z)^\top (x - z) dt.$$

Then,

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

Proof (2/2)

Proof of the theorem

We have shown that for all x,

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

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

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

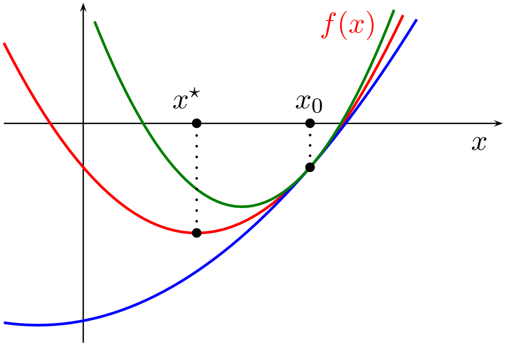
By summing from $t = 1$ to T, we have a telescopic sum

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

Introduction of a few optimization principles

An important inequality for smooth and μ-strongly convex functions

If ∇f is L-Lipschitz continuous and f μ-strongly convex



- $f(x) \leq f(x_0) + \nabla f(x_0)^\top (x - x_0) + \frac{L}{2}\|x - x_0\|_2^2$;
- $f(x) \geq f(x_0) + \nabla f(x_0)^\top (x - x_0) + \frac{\mu}{2}\|x - x_0\|_2^2$;

Introduction of a few optimization principles

Proposition

When f is μ-strongly convex and L-smooth, the gradient descent algorithm with step-size $1/L$ produces iterates such that

$$f(x_t) - f^\star \leq \left(1 - \frac{\mu}{L}\right)^t \frac{L\|x_0 - x^\star\|_2^2}{2}.$$

We call that a **linear** convergence rate.

Remarks

- if f is twice differentiable, L and μ represent the larget and smallest eigenvalues of the Hessian, respectively.
- L/μ is called the **condition number**.

Proof

We start from an inequality from the previous proof

$$\begin{aligned} f(x_t) &\leq f(x_{t-1}) + \nabla f(x_{t-1})^\top (x^\star - x_{t-1}) + \frac{L}{2}\|x^\star - x_{t-1}\|_2^2 - \frac{L}{2}\|x^\star - x_t\|_2^2 \\ &\leq f^\star + \frac{L-\mu}{2}\|x^\star - x_{t-1}\|_2^2 - \frac{L}{2}\|x^\star - x_t\|_2^2. \end{aligned}$$

In addition, we have that $f(x_t) \geq f^\star + \frac{\mu}{2}\|x_t - x^\star\|_2^2$, and thus

$$\begin{aligned} \|x^\star - x_t\|_2^2 &\leq \frac{L-\mu}{L+\mu}\|x^\star - x_{t-1}\|_2^2 \\ &\leq \left(1 - \frac{\mu}{L}\right)\|x^\star - x_{t-1}\|_2^2. \end{aligned}$$

Finally,

$$\begin{aligned} f(x_t) - f^\star &\leq \frac{L}{2}\|x_t - x^\star\|_2^2 \\ &\leq \left(1 - \frac{\mu}{L}\right)^t \frac{L\|x^\star - x_0\|_2^2}{2} \end{aligned}$$

Introduction of a few optimization principles

Remark: with stepsize $1/L$, gradient descent may be interpreted as a **majorization-minimization** algorithm:

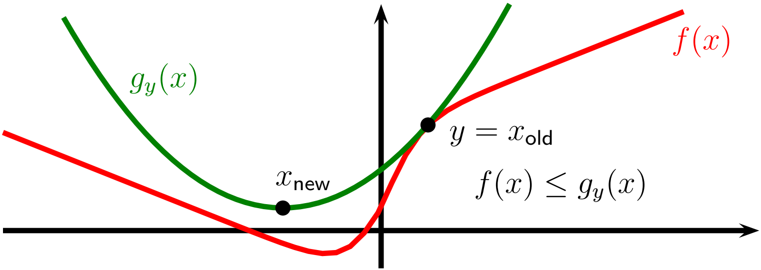
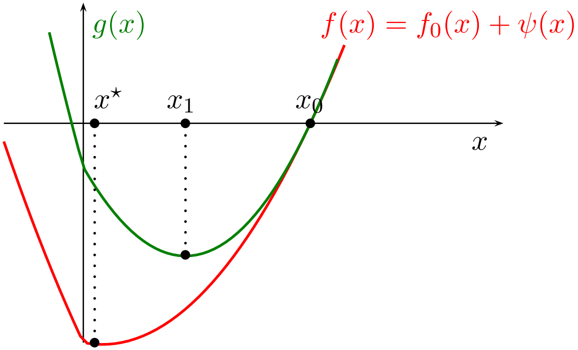


Figure: At each step, we update $x \in \arg\min_{x \in \mathbb{R}^p} g_y(x)$

The proximal gradient method

An important inequality for composite functions

If ∇f_0 is L-Lipschitz continuous



- $f_0(x) + \psi(x) \leq f_0(x_0) + \nabla f_0(x_0)^\top (x - x_0) + \frac{L}{2}\|x - x_0\|_2^2 + \psi(x)$;
- x_1 minimizes g.

The proximal gradient method

Gradient descent for minimizing f consists of

$$x_t \leftarrow \arg\min_{x \in \mathbb{R}^p} g_t(x) \quad \Longleftrightarrow \quad x_t \leftarrow x_{t-1} - \frac{1}{L}\nabla f(x_{t-1}).$$

The proximal gradient method for minimizing $f = f_0 + \psi$ consists of

$$x_t \leftarrow \arg\min_{x \in \mathbb{R}^p} g_t(x),$$

which is equivalent to

$$\boxed{x_t \leftarrow \arg\min_{x \in \mathbb{R}^p} \frac{1}{2}\left\| x_{t-1} - \frac{1}{L}\nabla f_0(x_{t-1}) - x \right\|_2^2 + \frac{1}{L}\psi(x).}$$

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

$$y \mapsto \arg\min_{x \in \mathbb{R}^p} \frac{1}{2}\|y - x\|_2^2 + \psi(x).$$

The proximal gradient method

Remarks

- also known as **forward-backward** algorithm;
- has similar convergence rates as the gradient descent method (the proof is nearly identical).
- there exists **line search schemes** to automatically tune L;

The case of ℓ_1

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

$$x[j] = \text{sign}(y[j])(|y[j]| - \lambda)^+.$$

The resulting algorithm is called **iterative soft-thresholding**.

[Nowak and Figueiredo, 2001, Daubechies et al., 2004, Combettes and Wajs, 2006, Beck and Teboulle, 2009, Wright et al., 2009, Nesterov, 2013]...

The proximal gradient method

The proximal operator for the group Lasso penalty

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

For $q = 2$,

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

For $q = \infty$,

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

These formula generalize soft-thresholding to groups of variables.

The proximal gradient method

A few proximal operators:

- ℓ_0-penalty: hard-thresholding;
- ℓ_1-norm: soft-thresholding;
- group-Lasso: group soft-thresholding;
- fused-lasso (1D total variation): [Hoefling, 2010];
- total variation: [Chambolle and Darbon, 2009];
- hierarchical norms: [Jenatton et al., 2011], $O(p)$ complexity;
- overlapping group Lasso with ℓ_∞-norm: [Mairal et al., 2010];

Accelerated gradient descent methods

Nesterov introduced in the 80's an acceleration scheme for the gradient descent algorithm. It was generalized later to the composite setting.

FISTA

$$x_t \leftarrow \arg\min_{x \in \mathbb{R}^p} \frac{1}{2} \left\| x - \left(y_{t-1} - \frac{1}{L} \nabla f_0(y_{t-1}) \right) \right\|_2^2 + \frac{1}{L}\psi(x);$$

$$\text{Find } \alpha_t > 0 \;\; \text{s.t.} \;\; \alpha_t^2 = (1-\alpha_t)\alpha_{t-1}^2 + \frac{\mu}{L}\alpha_t;$$

$$y_t \leftarrow x_t + \beta_t(x_t - x_{t-1}) \quad \text{with} \quad \beta_t = \frac{\alpha_{t-1}(1-\alpha_{t-1})}{\alpha_{t-1}^2 + \alpha_t}.$$

- $f(x_t) - f^\star = O(1/t^2)$ for **convex** problems;
- $f(x_t) - f^\star = O((1 - \sqrt{\mu/L})^t)$ for μ-**strongly convex** problems;
- Acceleration works in many practical cases.

see [Beck and Teboulle, 2009, Nesterov, 1983, 2004, 2013]

What do we mean by "acceleration"?

Complexity analysis for large finite sums

Since f is a sum of n functions, computing ∇f requires computing n gradients ∇f_i. The complexity to reach an $\varepsilon-$solution is given below

	$\mu > 0$	$\mu = 0$
ISTA	$O\left(n\frac{L}{\mu}\log\left(\frac{1}{\varepsilon}\right)\right)$	$O\left(\frac{nL}{\varepsilon}\right)$
FISTA	$O\left(n\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right)$	$O\left(n\sqrt{\frac{L}{\varepsilon}}\right)$

Remarks

- ε-solution means here $f(x_t) - f^\star \leq \varepsilon$.
- For $n = 1$, the rates of FISTA are optimal for a "first-order local black box" [Nesterov, 2004].
- For $L = 1$ and $\mu = 1/n$, scales at best in $n^{3/2}$.

How does "acceleration" work?

Unfortunately, the literature does not provide any simple geometric explanation...

How does "acceleration" work?

Unfortunately, the literature does not provide any simple geometric explanation... but they are a few obvious facts and a mechanism introduced by Nesterov, called **"estimate sequence"**.

Obvious fact

- Simple gradient descent steps are "blind" to the past iterates, and are based on a **purely local** model of the objective.
- Accelerated methods usually involve an **extrapolation step** $y_t = x_t + \beta_t(x_t - x_{t-1})$ with β_t in $(0, 1)$.
- Nesterov interprets acceleration as relying on a better model of the objective called **estimate sequence**.

How does "acceleration" work?

Definition of estimate sequence [Nesterov].

A pair of sequences $(\varphi_t)_{t\geq 0}$ and $(\lambda_t)_{t\geq 0}$, with $\lambda_t \geq 0$ and $\varphi_t : \mathbb{R}^p \to \mathbb{R}$, is called an **estimate sequence** of function f if $\lambda_t \to 0$ and

for any $x \in \mathbb{R}^p$ and all $t \geq 0$, $\quad \varphi_t(x) - f(x) \leq \lambda_t(\varphi_0(x) - f(x))$.

In addition, if for some sequence $(x_t)_{t\geq 0}$ we have

$$f(x_t) \leq \varphi_t^\star \triangleq \min_{x\in\mathbb{R}^p} \varphi_t(x),$$

then

$$f(x_t) - F^\star \leq \lambda_t(\varphi_0(x^\star) - f^\star),$$

where $x^\star$ is a minimizer of f.

How does “acceleration” work?

In summary, we need two properties

1. $\varphi_t(x) \leq (1 - \lambda_t)f(x) + \lambda_t \varphi_0(x)$;
2. $f(x_t) \leq \varphi_t^\star \triangleq \min_{x \in \mathbb{R}^p} \varphi_t(x)$.

Remarks

- φ_t is neither an upper-bound, nor a lower-bound;
- Finding the right estimate sequence is often nontrivial.

How does "acceleration" work?

In summary, we need two properties

1. $\varphi_t(x) \leq (1 - \lambda_t) f(x) + \lambda_t \varphi_0(x)$;
2. $f(x_t) \leq \varphi_t^\star \triangleq \min_{x \in \mathbb{R}^p} \varphi_t(x)$.

How to build an estimate sequence?

Define φ_t recursively

$$\varphi_t(x) \triangleq (1 - \alpha_t)\varphi_{t-1}(x) + \alpha_t d_t(x),$$

where d_t is a **lower-bound**, e.g., if F is smooth,

$$d_t(x) \triangleq F(y_t) + \nabla F(y_t)^\top (x - y_t) + \frac{\mu}{2}\|x - y_t\|_2^2,$$

Then, work hard to choose α_t as large as possible, and y_t and x_t such that property 2 holds. Subsequently, $\lambda_t = \prod_{t=1}^{t}(1 - \alpha_t)$.

The stochastic (sub)gradient descent algorithm

Consider now the minimization of an expectation

$$\min_{x \in \mathbb{R}^p} f(x) = \mathbb{E}_z[\ell(x, z)],$$

To simplify, we assume that for all z, $x \mapsto \ell(x, z)$ is differentiable.

Algorithm

At iteration t,

- Randomly draw one example z_t from the training set;
- Update the current iterate

$$x_t \leftarrow x_{t-1} - \eta_t \nabla f_t(x_{t-1}) \quad \text{with} \quad f_t(x) = \ell(x, z_t).$$

- Perform online averaging of the iterates (optional)

$$\tilde{x}_t \leftarrow (1 - \gamma_t)\tilde{x}_{t-1} + \gamma_t x_t.$$

The stochastic (sub)gradient descent algorithm

There are various learning rates strategies (constant, varying step-sizes), and averaging strategies. Depending on the problem assumptions and choice of η_t, γ_t, classical convergence rates may be obtained:

- $f(\tilde{x}_t) - f^\star = O(1/\sqrt{t})$ for convex problems;
- $f(\tilde{x}_t) - f^\star = O(1/t)$ for strongly-convex ones;

Remarks

- The convergence rates are not great, but the complexity **per-iteration** is small (1 gradient evaluation for minimizing an empirical risk versus n for the batch algorithm).
- When the amount of data is infinite, the method **minimizes the expected risk** (which is what we want).
- Choosing a good learning rate automatically is an open problem.

Proof of an $O(1/\sqrt{t})$ rate for the convex case

Inspired by (aka, stolen from) F. Bach's slides

Assumptions

- The solution lies in a bounded domain $\mathcal{C} = \{\|x\| \leq D\}$.
- The sub-gradients are bounded on $\mathcal{C}$: $\|\nabla f_t(x)\| \leq B$.
- Fix in advance the number of iterations T and choose $\eta_t = \frac{2D}{B\sqrt{T}}$.
- Choose Polyak-Ruppert averaging $\tilde{x}_T = (1/T) \sum_{t=0}^{T-1} x_t$.
- Perform updates with projections

$$x_t \leftarrow \Pi_{\mathcal{C}}[x_{t-1} - \eta_t \nabla f_t(x_{t-1})].$$

Proposition

$$\mathbb{E}[f(\tilde{x}_t) - f^\star] \leq \frac{2DB}{\sqrt{T}}.$$

Proof of an $O(1/\sqrt{t})$ rate for the convex case

Inspired by (aka, stolen from) F. Bach's slides

- $\mathcal{F}_t$: information up to time t.
- $\|x\| \leq D$ and $\|\nabla f_t(x)\| \leq B$. Besides $\mathbb{E}[\nabla f_t(x)|\mathcal{F}_{t-1}] = \nabla f(x)$.

$$\begin{aligned}\|x_t - x^\star\|^2 &\leq \|x_{t-1} - \eta_t \nabla f_t(x_{t-1}) - x^\star\|^2 \\ &\leq \|x_{t-1} - x^\star\|^2 + B^2\eta_t^2 - 2\eta_t(x_{t-1} - x^\star)^\top \nabla f_t(x_{t-1}).\end{aligned}$$

Take now **conditional expectations**

$$\begin{aligned}\mathbb{E}[\|x_t - x^\star\|^2|\mathcal{F}_{t-1}] &\leq \|x_{t-1} - x^\star\|^2 + B^2\eta_t^2 - 2\eta_t(x_{t-1} - x^\star)^\top \nabla f(x_{t-1}) \\ &\leq \|x_{t-1} - x^\star\|^2 + B^2\eta_t^2 - 2\eta_t(f(x_{t-1}) - f^\star).\end{aligned}$$

Take now **full expectations**

$$\mathbb{E}[\|x_t - x^\star\|^2] \leq \mathbb{E}[\|x_{t-1} - x^\star\|^2] + B^2\eta_t^2 - 2\eta_t\mathbb{E}[f(x_{t-1}) - f^\star],$$

and, after reorganizing the terms

$$\mathbb{E}[f(x_{t-1}) - f^\star] \leq \frac{B^2\eta_t^2}{2} + \frac{1}{2\eta_t}\left(\mathbb{E}[\|x_{t-1} - x^\star\|^2] - \mathbb{E}[\|x_t - x^\star\|^2]\right).$$

Proof of an $O(1/\sqrt{t})$ rate for the convex case

Inspired by (aka, stolen from) F. Bach's slides

We start again from

$$\mathbb{E}[f(x_{t-1}) - f^\star] \leq \frac{B^2\eta_t^2}{2} + \frac{1}{2\eta_t}\left(\mathbb{E}[\|x_{t-1} - x^\star\|^2] - \mathbb{E}[\|x_t - x^\star\|^2]\right).$$

and we exploit the telescopic sum

$$\sum_{t=1}^{T}\mathbb{E}[f(x_{t-1}) - f^\star] \leq \sum_{t=1}^{T}\frac{B^2\eta_t^2}{2} + \sum_{t=1}^{T}\frac{1}{2\eta_t}\left(\mathbb{E}[\|x_{t-1} - x^\star\|^2] - \mathbb{E}[\|x_t - x^\star\|^2]\right)$$

$$\leq T\frac{B^2\eta^2}{2} + \frac{4D^2}{2\eta} \leq 2DB\sqrt{T} \quad \text{with} \quad \gamma = \frac{2D}{B\sqrt{T}}.$$

Finally, we conclude by using a convexity inequality

$$\mathbb{E}f\left(\frac{1}{T}\sum_{t=0}^{T-1}\right) - f^\star \leq \frac{2DB}{\sqrt{T}}.$$

Back to finite sums

Consider now the case of interest for us today:

$$\min_{x \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} f_i(x),$$

Question

Can we do as well as SGD in terms of cost per iteration, while enjoying a fast (linear) convergence rate like (accelerated or not) gradient descent?

For $n = 1$, no!

The rates are optimal for a "first-order local black box" [Nesterov, 2004].

For $n \geq 1$, yes! We need to design algorithms

- whose per-iteration **computational complexity** is smaller than n;
- whose **convergence rate** may be worse than FISTA....
- ...but with a better expected **computational complexity**.

Incremental gradient descent methods

$$\min_{x \in \mathbb{R}^p} \left\{ f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x) \right\}.$$

Several **randomized** algorithms are designed with one ∇f_i computed per iteration, with **fast convergence rates**, e.g., SAG [Schmidt et al., 2013]:

$$x_k \leftarrow x_{k-1} - \frac{\gamma}{Ln} \sum_{i=1}^{n} y_i^k \quad \text{with} \quad y_i^k = \begin{cases} \nabla f_i(x_{k-1}) & \text{if} \quad i = i_k \\ y_i^{k-1} & \text{otherwise} \end{cases}.$$

Incremental gradient descent methods

$$\min_{x\in\mathbb{R}^p}\left\{f(x)=\frac{1}{n}\sum_{i=1}^{n}f_i(x)\right\}.$$

Several **randomized** algorithms are designed with one ∇f_i computed per iteration, with **fast convergence rates**, e.g., SAG [Schmidt et al., 2013]:

$$x_k \leftarrow x_{k-1}-\frac{\gamma}{Ln}\sum_{i=1}^{n}y_i^k \quad \text{with} \quad y_i^k=\begin{cases}\nabla f_i(x_{k-1}) & \text{if } \; i=i_k\\ y_i^{k-1} & \text{otherwise}\end{cases}.$$

See also SVRG, SAGA, SDCA, MISO, Finito...
Some of these algorithms perform updates of the form

$$x_k \leftarrow x_{k-1}-\eta_k g_k \quad \text{with} \quad \mathbb{E}[g_k]=\nabla f(x_{k-1}),$$

but g_k has **lower variance** than in SGD.

[Schmidt et al., 2013, Xiao and Zhang, 2014, Defazio et al., 2014a,b, Shalev-Shwartz and Zhang, 2012, Mairal, 2015, Zhang and Xiao, 2015]

Incremental gradient descent methods

These methods achieve low **(worst-case)** complexity in expectation.
The number of gradients evaluations to ensure $f(x_k) - f^\star \leq \varepsilon$ is

	$\mu > 0$
FISTA	$O\left(n\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right)$
SVRG, SAG, SAGA, SDCA, MISO, Finito	$O\left(\max\left(n, \frac{\bar{L}}{\mu}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$

Incremental gradient descent methods

These methods achieve low **(worst-case)** complexity in expectation.
The number of gradients evaluations to ensure $f(x_k) - f^\star \leq \varepsilon$ is

	$\mu > 0$
FISTA	$O\left(n\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right)$
SVRG, SAG, SAGA, SDCA, MISO, Finito	$O\left(\max\left(n, \frac{\bar{L}}{\mu}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$

Main features vs. stochastic gradient descent

- Same complexity per-iteration (but higher memory footprint).
- **Faster convergence** (exploit the finite-sum structure).
- **Less parameter tuning** than SGD.
- Some variants are **compatible with a composite term** ψ.

Incremental gradient descent methods

These methods achieve low **(worst-case)** complexity in expectation.
The number of gradients evaluations to ensure $f(x_k) - f^\star \leq \varepsilon$ is

	$\mu > 0$
FISTA	$O\left(n\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right)$
SVRG, SAG, SAGA, SDCA, MISO, Finito	$O\left(\max\left(n, \frac{\bar{L}}{\mu}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$

Important remarks

- When $f_i(x) = \ell(z_i^\top x)$, the memory footprint is $O(n)$ otherwise $O(dn)$, except for SVRG ($O(d)$).
- Some algorithms require an estimate of μ;
- $\bar{L}$ is the average (or max) of the Lipschitz constants of the ∇f_i's.
- The L for fista is the Lipschitz constant of ∇f: $L \leq \bar{L}$.

Incremental gradient descent methods

stealing again a bit from F. Bach's slides.

Variance reduction

Consider two random variables X, Y and define

$$Z = X - Y + \mathbb{E}[Y].$$

Then,

- $\mathbb{E}[Z] = \mathbb{E}[X]$
- $\text{Var}(Z) = \text{Var}(X) + \text{Var}(Y) - 2\text{cov}(X, Y)$.

The variance of Z may be smaller if X and Y are positively correlated.

Incremental gradient descent methods

stealing again a bit from F. Bach's slides.

Variance reduction

Consider two random variables X, Y and define

$$Z = X - Y + \mathbb{E}[Y].$$

Then,

- $\mathbb{E}[Z] = \mathbb{E}[X]$
- $\text{Var}(Z) = \text{Var}(X) + \text{Var}(Y) - 2\text{cov}(X, Y)$.

The variance of Z may be smaller if X and Y are positively correlated.

Why is it useful for stochastic optimization?

- step-sizes for SGD have to decrease to ensure convergence.
- with variance reduction, one may use constant step-sizes.

Incremental gradient descent methods

SVRG

$$x_t = x_{t-1} - \gamma \left(\nabla f_{i_t}(x_{t-1}) - \nabla f_{i_t}(y) + \nabla f(y) \right),$$

where y is updated every epoch and $\mathbb{E}[\nabla f_{i_t}(y)|\mathcal{F}_{t-1}] = \nabla f(y)$.

SAGA

$$x_t = x_{t-1} - \gamma \left(\nabla f_{i_t}(x_{t-1}) - y_{i_t}^{t-1} + \frac{1}{n}\sum_{i=1}^n y_i^{t-1} \right),$$

where $\mathbb{E}[y_{i_t}^{t-1}|\mathcal{F}_{t-1}] = \frac{1}{n}\sum_{i=1}^n y_i^{t-1}$ and $y_i^t = \begin{cases} \nabla f_i(x_{t-1}) & \text{if } i = i_t \\ y_i^{t-1} & \text{otherwise.} \end{cases}$

MISO/Finito: for $n \geq L/\mu$, same form as SAGA but

$$\frac{1}{n}\sum_{i=1}^n y_i^{t-1} = -\mu x_{t-1} \quad \text{and} \quad y_i^t = \begin{cases} \nabla f_i(x_{t-1}) - \mu x_{t-1} & \text{if } i = i_t \\ y_i^{t-1} & \text{otherwise.} \end{cases}$$

Can we do even better for large finite sums?

Without vs with acceleration

	$\mu > 0$
FISTA	$O\left(n\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right)$
SVRG, SAG, SAGA, SDCA, MISO, Finito	$O\left(\max\left(n, \frac{\bar{L}}{\mu}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$
Accelerated versions	$\tilde{O}\left(\max\left(n, \sqrt{n\frac{\bar{L}}{\mu}}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$

- Acceleration for specific algorithms [Shalev-Shwartz and Zhang, 2014, Lan, 2015, Allen-Zhu, 2016].
- Generic acceleration: Catalyst [Lin et al., 2015].
- see [Agarwal and Bottou, 2015] for discussions about optimality.

What we have not (or should have) covered

Import approaches and concepts

- distributed optimization.
- proximal splitting / ADMM.
- Quasi-Newton approaches.

What we have not (or should have) covered

Import approaches and concepts

- distributed optimization.
- proximal splitting / ADMM.
- Quasi-Newton approaches.

The question

Should we care that much about minimizing finite sums when all we want is minimizing an expectation?

Part II: Sparse estimation

Chronological overview of parsimony

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

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

How to choose k?

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

These approaches lead to penalized problems

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

with different choices of λ depending on the chosen criterion.

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

How to solve the best k-subset selection problem?

Unfortunately...

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

Two strategies

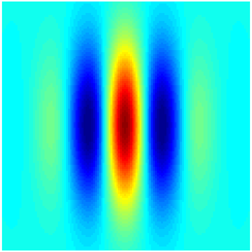
- **combinatorial exploration** with branch-and-bound techniques [Furnival and Wilson, 1974] $\rightarrow$ **leaps and bounds**, exact algorithm but exponential complexity;
- **greedy approach**: forward selection [Efroymson, 1960] (originally developed for observing *intermediate* solutions), already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

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

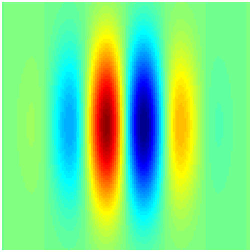
Wavelet thresholding in signal processing from the 90's

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

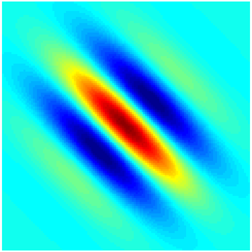
- 2D-Gabors [Daugman, 1985];
- steerable wavelets [Simoncelli et al., 1992];
- curvelets [Candès and Donoho, 2002];
- countourlets [Do and Vertterli, 2003];
- bandlets [Le Pennec and Mallat, 2005];
- $\star$-lets (joke).



(a) 2D Gabor filter.



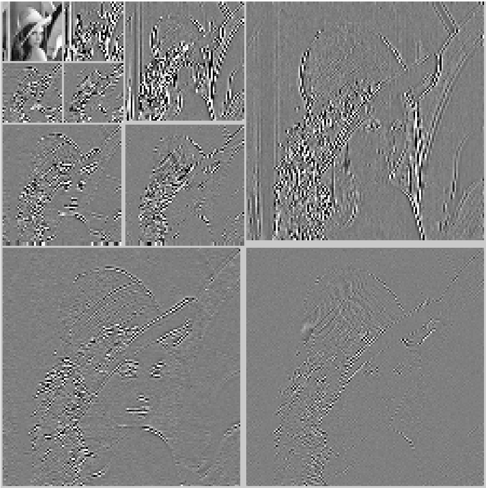
(b) With shifted phase.



(c) With rotation.

Wavelet thresholding in signal processing from 90's

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



Wavelet thresholding in signal processing from 90's

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

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

The k-sparse approximation problem

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

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

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

Wavelet thresholding in signal processing from 90's

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

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

The k-sparse approximation problem

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

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

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

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

Wavelet thresholding in signal processing, 90's

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

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

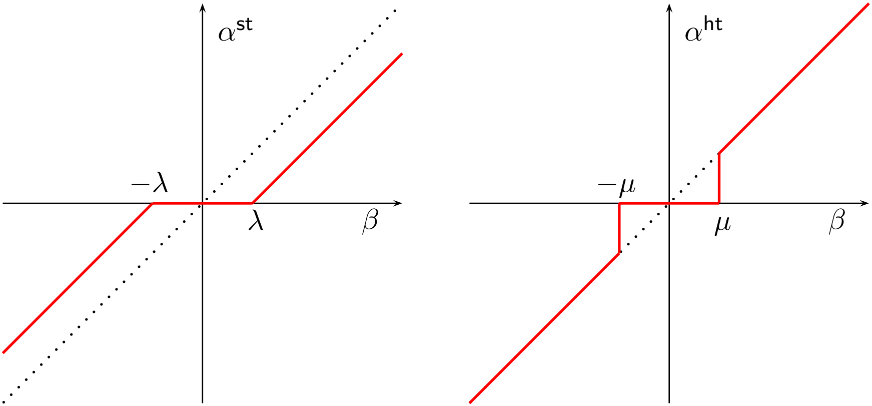
where λ is a parameter playing the same role as μ previously.

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

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

which will be of high importance later.

Wavelet thresholding in signal processing, 90's



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

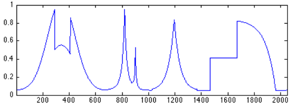
(e) Hard-thresholding operator $\alpha^{\text{ht}} = \delta_{|\beta| \geq \mu}\beta$.

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

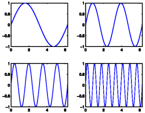
The modern parsimony and the ℓ_1-norm

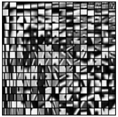
Sparse linear models in signal processing

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



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





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

$$\underbrace{\begin{pmatrix} \\ \mathbf{x} \\ \\ \end{pmatrix}}_{\mathbf{x} \in \mathbb{R}^n} \approx \underbrace{\left(\begin{array}{c|c|c|c} & & & \\ \mathbf{d}_1 & \mathbf{d}_2 & \cdots & \mathbf{d}_p \\ & & & \end{array}\right)}_{\mathbf{D} \in \mathbb{R}^{n \times p}} \underbrace{\begin{pmatrix} \boldsymbol{\alpha}[1] \\ \boldsymbol{\alpha}[2] \\ \vdots \\ \boldsymbol{\alpha}[p] \end{pmatrix}}$$

The modern parsimony and the ℓ_1-norm

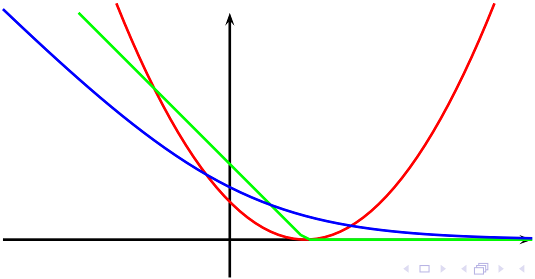
Sparse linear models: machine learning/statistics point of view

A few examples:

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

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

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



The modern parsimony and the ℓ_1-norm

Sparse linear models: machine learning/statistics point of view

A few examples:

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

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

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

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

The modern parsimony and the ℓ_1-norm

Why does the ℓ_1-norm induce sparsity?

Can we get some intuition from the simplest isotropic case?

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

or equivalently the Euclidean projection onto the ℓ_1-ball?

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

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

The modern parsimony and the ℓ_1-norm

Why does the ℓ_1-norm induce sparsity?

Can we get some intuition from the simplest isotropic case?

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

or equivalently the Euclidean projection onto the ℓ_1-ball?

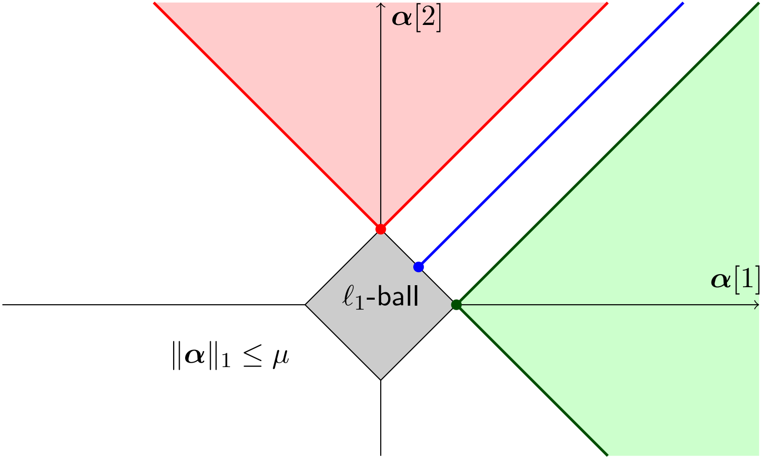
$$\tilde{\boldsymbol{\alpha}}(\mu) = \arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{x} - \boldsymbol{\alpha}\|_2^2 \;\text{ s.t. }\; \|\boldsymbol{\alpha}\|_1 \leq \mu.$$

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

The relation between μ and λ is unknown a priori.

Why does the ℓ_1-norm induce sparsity?

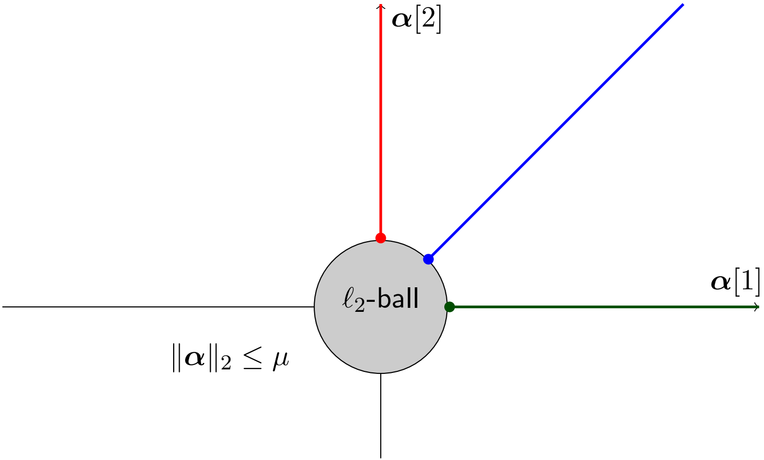
Regularizing with the ℓ_1-norm



The projection onto a convex set is "biased" towards singularities.

Why does the ℓ_1-norm induce sparsity?

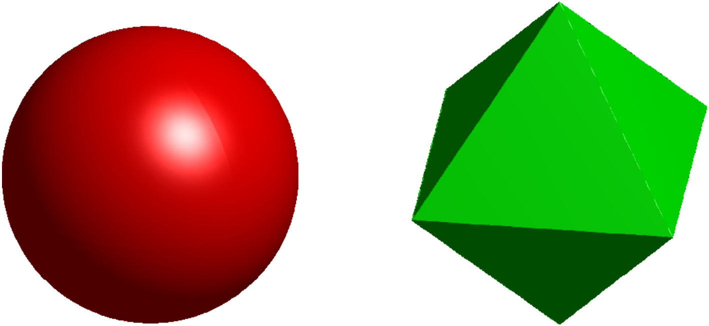
Regularizing with the ℓ_2-norm



The ℓ_2-norm is isotropic.

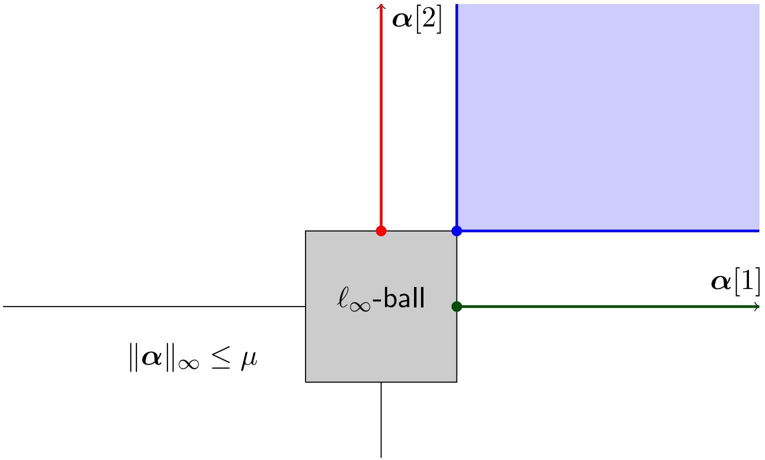
Why does the ℓ_1-norm induce sparsity?

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



Why does the ℓ_1-norm induce sparsity?

Regularizing with the ℓ_∞-norm



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

Why does the ℓ_1-norm induce sparsity?

Analytical point of view: 1D case

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

Piecewise quadratic function with a kink at zero.

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

Optimality conditions. α is optimal iff:

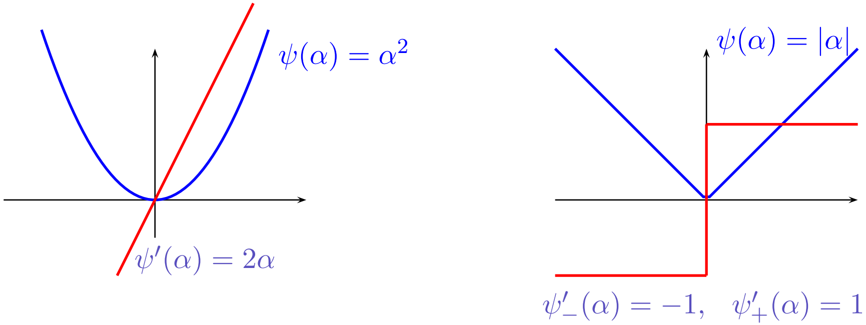
- $|\alpha| > 0$ and $(x - \alpha) + \lambda \operatorname{sign}(\alpha) = 0$
- $\alpha = 0$ and $g_+ \geq 0$ and $g_- \leq 0$

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

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

Why does the ℓ_1-norm induce sparsity?

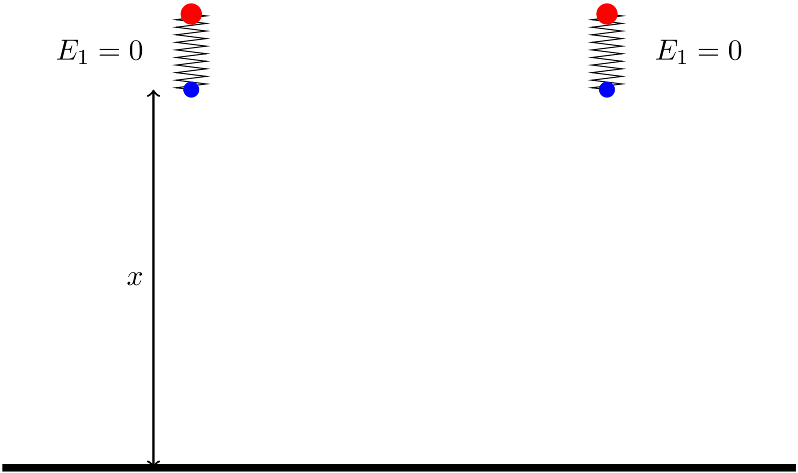
Comparison with ℓ_2-regularization in 1D



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

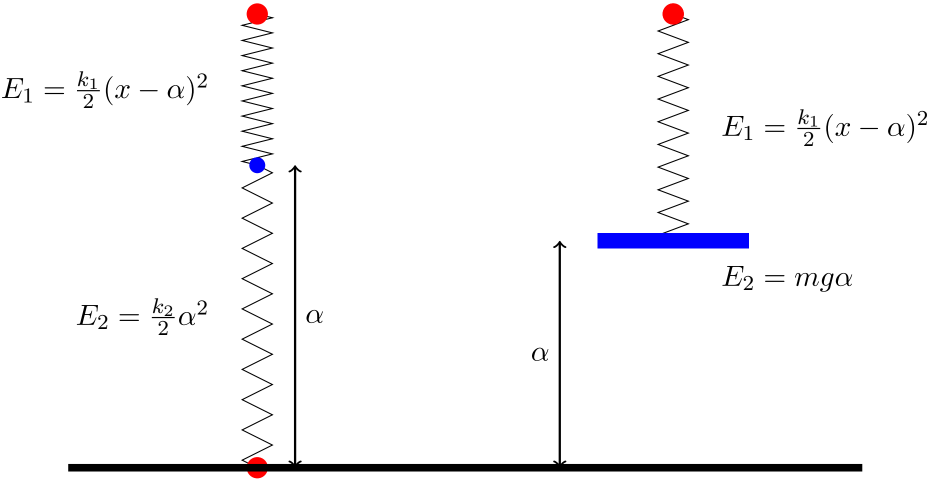
Why does the ℓ_1-norm induce sparsity?

Physical illustration



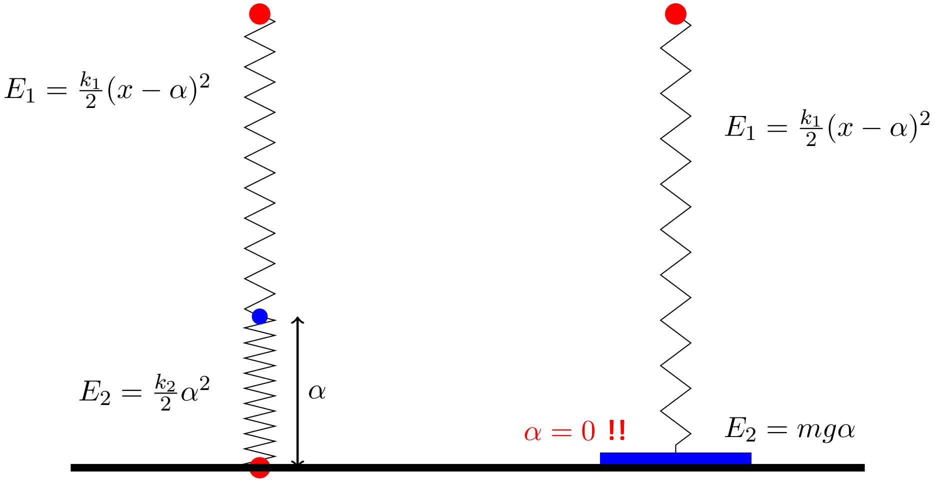
Why does the ℓ_1-norm induce sparsity?

Physical illustration



Why does the ℓ_1-norm induce sparsity?

Physical illustration



Why does the ℓ_1-norm induce sparsity?

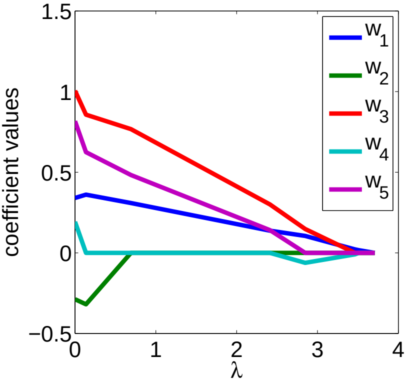


Figure: The regularization path of the Lasso.

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

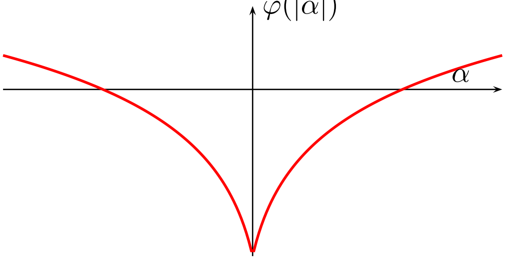
Non-convex sparsity-inducing penalties

Exploiting concave functions with a kink at zero

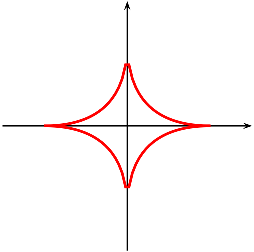
$\psi(\boldsymbol{\alpha}) = \sum_{j=1}^{p} \varphi(|\boldsymbol{\alpha}[j]|)$.

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

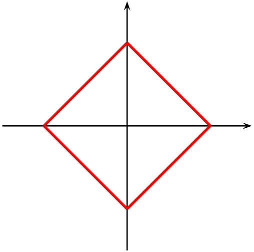
φ is any function that looks like this:



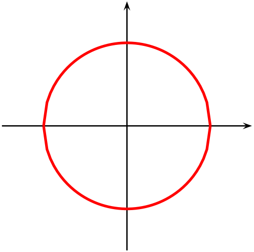
Non-convex sparsity-inducing penalties



(a) $\ell_{0.5}$-ball, 2-D



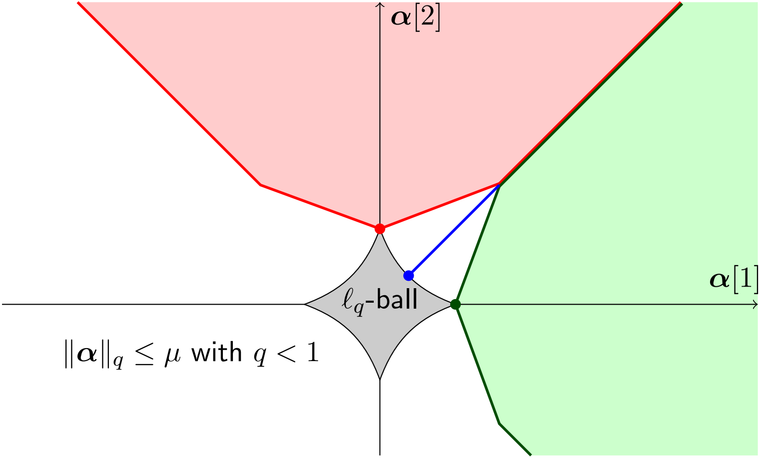
(b) ℓ_1-ball, 2-D



(c) ℓ_2-ball, 2-D

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

Non-convex sparsity-inducing penalties

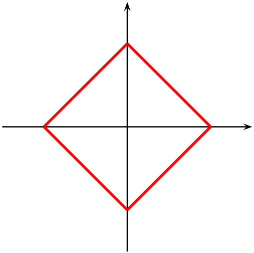


Elastic-net

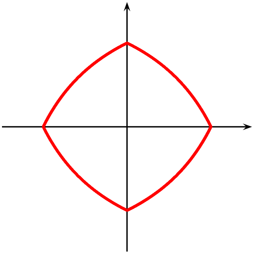
The **elastic net** introduced by [Zou and Hastie, 2005]

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

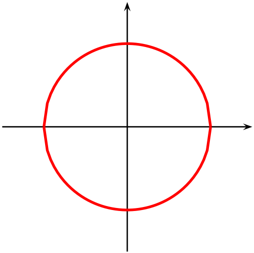
The penalty provides more stable (but less sparse) solutions.



(a) ℓ_1-ball, 2-D



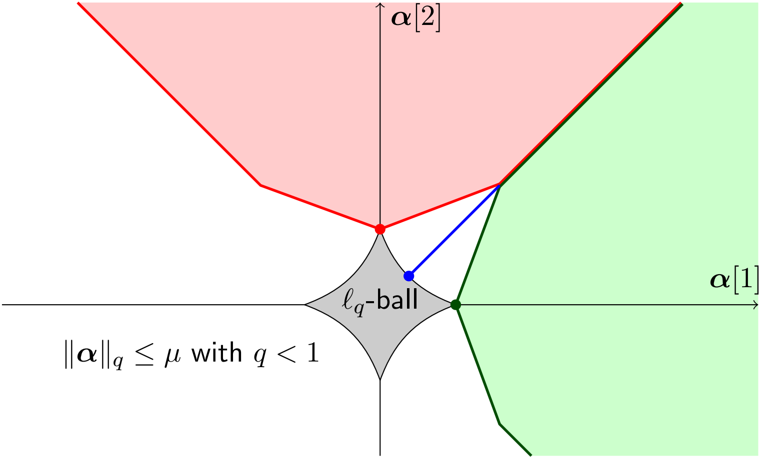
(b) elastic-net, 2-D



(c) ℓ_2-ball, 2-D

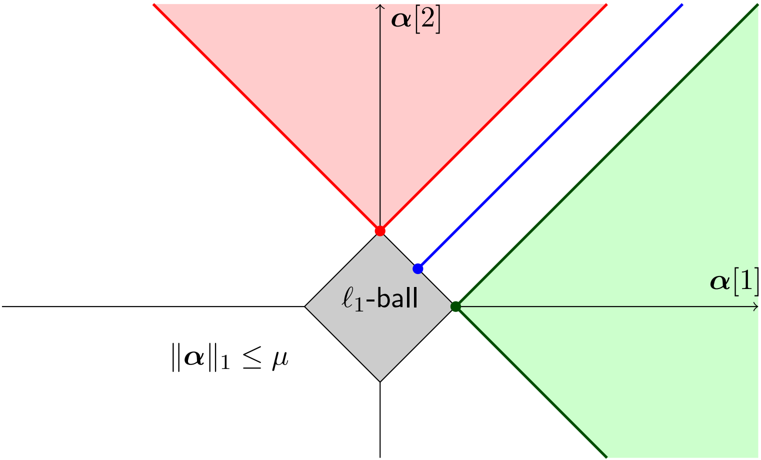
The elastic-net

vs other penalties



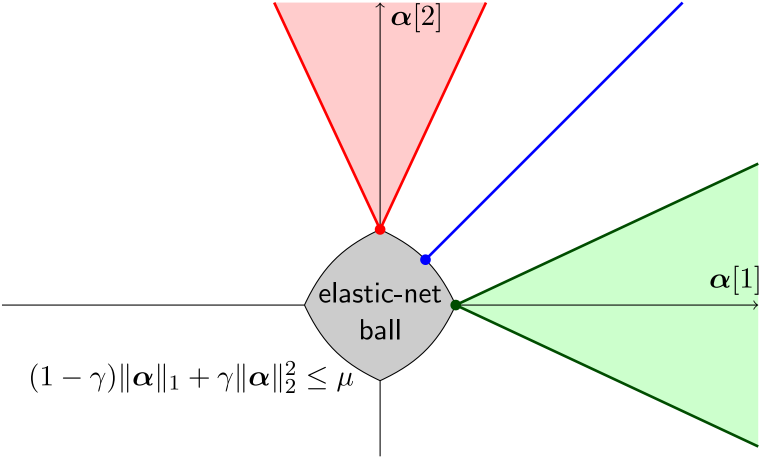
The elastic-net

vs other penalties



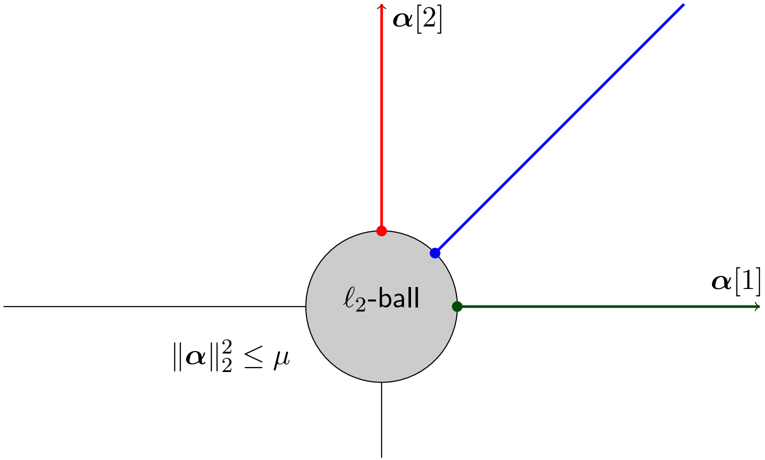
The elastic-net

vs other penalties



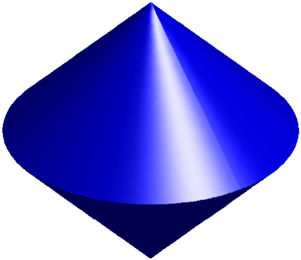
The elastic-net

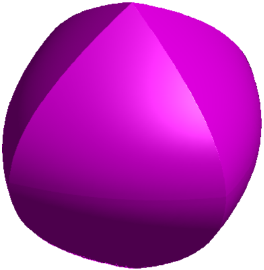
vs other penalties



Structured sparsity

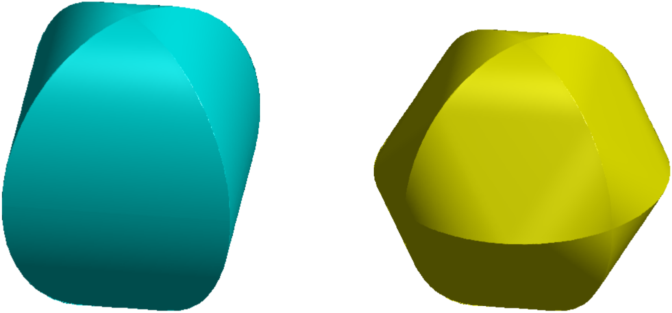
images produced by G. Obozinski





Structured sparsity

images produced by G. Obozinski



Mark the date! July 2-6th, Grenoble

Along with Naver Labs, Inria is organizing a summer school in Grenoble on artificial intelligence. Visit https://project.inria.fr/paiss/.

Among the distinguished speakers

- Lourdes Agapito (UCL)
- Kyunghyun Cho (NYU/Facebook)
- Emmanuel Dupoux (EHESS)
- Martial Hebert (CMU)
- Hugo Larochelle (Google Brain)
- Yann LeCun (Facebook/NYU)
- Jean Ponce (Inria)
- Cordelia Schmid (Inria)
- Andrew Zisserman (Oxford/Google DeepMind).
- ...

References I

A. Agarwal and L. Bottou. A lower bound for the optimization of finite sums. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2015.

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

Zeyuan Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *arXiv preprint arXiv:1603.05953*, 2016.

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.

Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X Charles, D Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. *The Journal of Machine Learning Research*, 14 (1):3207–3260, 2013.

References II

Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *arXiv preprint arXiv:1606.04838*, 2016.

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

Antonin Chambolle and Jérôme Darbon. On total variation minimization and surface evolution using parametric maximum flows. *International journal of computer vision*, 84(3):288, 2009.

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

P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *SIAM Multiscale Modeling and Simulation*, 4(4):1168–1200, 2006.

David Corfield, Bernhard Schölkopf, and Vladimir Vapnik. Falsificationism and statistical learning theory: Comparing the popper and vapnik-chervonenkis dimensions. *Journal for General Philosophy of Science*, 40(1):51–58, 2009.

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

References III

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

A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems (NIPS)*, 2014a.

A. J. Defazio, T. S. Caetano, and J. Domke. Finito: A faster, permutable incremental gradient method for big data problems. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2014b.

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

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

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

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

References IV

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

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

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

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

Guanghui Lan. An optimal randomized incremental gradient method. *arXiv preprint arXiv:1507.02000*, 2015.

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

H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, 2015.

References V

J. Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2): 829–855, 2015.

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

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

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

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

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

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

References VI

Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/k2). In *Doklady an SSSR*, volume 269, pages 543–547, 1983.

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

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

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

M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *arXiv:1309.2388*, 2013.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. *arXiv preprint arXiv:1206.6471*, 2012.

References VII

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

S. Shalev-Shwartz and T. Zhang. Proximal stochastic dual coordinate ascent. *arXiv:1211.2717*, 2012.

S. Shalev-Shwartz and T. Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, pages 1–41, 2014.

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

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

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1995.

S.J. Wright, R.D. Nowak, and M.A.T. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7): 2479–2493, 2009.

References VIII

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

L. Xiao and T. Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

Y. Zhang and L. Xiao. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2015.

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