OPTIMIZATION FOR MACHINE LEARNING

Regularized, large-scale and distributed optimization

November 16, 2023

Today: Sparsity and LASSO

SPARSITY AND REGULARIZATION

Motivation: Sparse models, typically because of overparametrization

(Lots of) zeros

$\begin{bmatrix}1\\0\end{bmatrix} \in \mathbb{R}^2$ Sparse?

Yes, 50% of the coefficients are 0

$\begin{bmatrix}1 & & 0\\ & \ddots & \\ 0 & & 1\end{bmatrix} \in \mathbb{R}^{10^9 \times 10^{12}}$ Sparse!

$\frac{10^{9+12}-1}{10^{9+12}} \times 100$ zero coefficients

① Sparse regularizers

Recall:

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \ \underbrace{f(x)}_{\text{data-fitting term}} + \overset{\lambda > 0}{\lambda} \underbrace{\Omega(x)}_{\text{regularization term / regularizer}}$$

Q) What regularizer can we use to produce solutions that are sparser (more zero coefficients) than the solutions of

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \ f(x) \quad ? \qquad \text{(un-regularized problem)}$$

Natural choice: "ℓ_0 norm" (read "ell zero")

Def: The ℓ_0-norm in $\mathbb{R}^d$ is the function $\|\cdot\|_0 : \mathbb{R}^d \to \mathbb{R}$ defined by

$$\forall x \in \mathbb{R}^d, \quad \|x\|_0 = \sum_{j=1}^{d} \mathbb{1}(x_j \neq 0)$$

$x = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}$

$\mathbb{1}(x_j \neq 0) = \begin{cases} 1 & \text{if } x_j \neq 0 \\ 0 & \text{otherwise} \end{cases}$

→ $\|x\|_0$ is the number of nonzero coordinates in x

→ $\|x\|_0 \in \{0, 1, \dots, d\}$

→ $\forall (x,y) \in (\mathbb{R}^d)^2$, x is sparser than y if $\|x\|_0 < \|y\|_0$

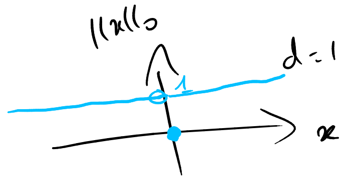
↳ If we consider

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \; f(x) + \lambda \|x\|_0 ,$$

then the regularization term penalizes the vectors with the largest values of $\|x\|_0$

$$\lambda \to \infty \qquad \underset{x \in \mathbb{R}^d}{\text{minimize}} \; \|x\|_0 \;\to\; x^* = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \to \; \min_{x \in \mathbb{R}^d} \|x\|_0 = 0$$

↳ Issues:



Key optimization challenges

- $\|\cdot\|_0$ is nonconvex
- $\|\cdot\|_0$ is discontinuous (big issue in optimization)

Additional challenges

- $\|\cdot\|_0$ has a combinatorial structure
- $\|\cdot\|_0$ is not even a norm

↳ In practice, we use regularizers that approximate the ℓ_0 norm and are easier to use in an optimization problem.

- The ℓ_0 norm is a limiting case of a family of functions called the "ℓ_p norms"

$$\forall p \in [0, +\infty], \qquad \|x\|_p := \begin{cases} \sum_{j=1}^{d} \mathbb{1}(x_j \neq 0) & \text{if } p = 0 \\ \max_{1 \le j \le d} |x_j| & \text{if } p = \infty \\ \left(\sum_{j=1}^{d} |x_j|^p \right)^{1/p} & \text{if } 0 < p < \infty \end{cases}$$

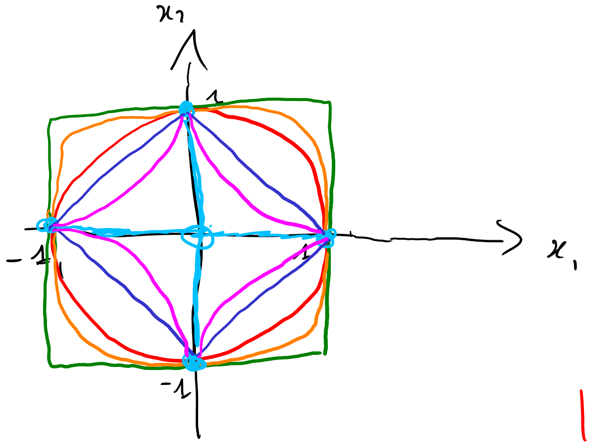
$p=2$ $\qquad \|x\|_2 = \|x\| = \sqrt{\sum_{j=1}^{d} x_j^2}$

$p=1$ $\qquad \|x\|_1 = \sum_{j=1}^{d} |x_j|$

$$\|x\|_p \xrightarrow[\substack{p \to 0 \\ p > 0}]{} \|x\|_0$$

$$\|x\|_p \xrightarrow[p \to \infty]{} \|x\|_\infty$$

$d=2$



$\|x\|_2 = 1$ $\qquad$ $\|x\|_4 = 1$

$\|x\|_\infty = 1$ $\qquad$ $\|x\|_0 = 1$

$\|x\|_1 = 1$ $\qquad$ $\|x\|_{1/2} = 1$

- $\|.\|_p$ is a norm when $p \geq 1$
- $\|.\|_p$ is a convex function when $p \geq 1$

$\Rightarrow$ suggest that we could use $\|.\|_p$ with $p \geq 1$ to approximate $\|.\|_0$

<u>Key fact</u>: $\|.\|_1$ is the best convex upper bound of the $\|.\|_0$ norm, i.e.

$\forall$ convex function $g : \mathbb{R}^d \to \mathbb{R}$ such that

$$g(x) \geq \|x\|_0 \quad \forall x \in \mathbb{R}^d,$$

Then $\qquad g(x) \geq \|x\|_1 \quad \forall x \in \mathbb{R}^d$

(Using $\|.\|_p$ with $0 < p < 1$ yields a nonconvex upper bound)

↳ the most popular choice for a sparse regularizer is the ℓ_1 norm $\Omega(x) = \|x\|_1$, also called the LASSO regularization term (typically when $f(x) = \frac{1}{2}\| - \|_2^2$)

⇒ there are numerous variations on this simple choice :

Group LASSO (aka ℓ_1/ℓ_2 regularizer)

$$\Omega(x) = \sum_{g \in G} \|x_g\|_2 \quad \text{where } G \text{ is a partition of } \{1, \dots, d\}$$

$x_g = [x_j]_{j \in g}$

(Ex $\underbrace{1 \quad \frac{d}{2}}\underbrace{\frac{d}{2}+1 \quad d}$)

- Groups of coordinates $g \in G$
- Models the a priori that a given group of parameters should either be used (all non zeros) or not used (zeros)

$G = \{\{1\}, \{2\}, \dots, \{d\}\} \Rightarrow \|\cdot\|_1$ norm

$\sum_{g \in G} \|x_g\|_2 = \|y\|_1$ where $y = \big[\|x_g\|_2\big]_{g \in G} \in \mathbb{R}^{|G|}$

[$G = \{\{1, \dots, d\}\} \Rightarrow \sum_{g \in G} \|x_g\|_2 = \|x\|_2$

__Remark__: The group regularizers (ℓ_1/ℓ_2) are often use to encode links between the parameters, which is problem-specific.

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad f(x) + \lambda \underbrace{\|x\|_1}$$

penalizes vectors that have nonzero coordinates

$G = \{\{1\}, \{2, \dots, d\}\}$

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad f(x) + \lambda \underbrace{\left(|x_1| + \left\| \begin{bmatrix} x_2 \\ \vdots \\ x_d \end{bmatrix} \right\|_2 \right)}$$

penalizes vectors that either have nonzero first coordinate or that have a nonzero norm for $\begin{bmatrix} x_2 \\ \vdots \\ x_d \end{bmatrix}$

$$|x_1| + \sqrt{x_2^2 + \dots + x_d^2}$$

$$\neq |x_1| + |x_2| + \dots + |x_d|$$

→ this idea generalizes further:

- Can replace the ℓ_2 norm by an ℓ_q norm with $q > 1$

  (ex) $\quad \Omega(x) = \sum_{g \in G} \|x_g\|_\infty$

- Can use overlapping groups

$$\Omega(x) = \|x\|_2 + |x_1| + |x_2|$$

② The case of ℓ_1 regularization

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad f(x) + \lambda \|x\|_1 \qquad \lambda > 0$$

Special case : LASSO estimator (1990s)

$A \in \mathbb{R}^{n \times d}$, $y \in \mathbb{R}^n$

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad \frac{1}{2n} \|Ax - y\|_2^2 + \lambda \|x\|_1$$

↳ First approach : $\|\cdot\|_1$ is nonsmooth (its gradient does not exist at certain points)

$\Rightarrow$ Fix : Use subgradients!

↳ Second approach : $x \mapsto \frac{1}{2n} \|Ax - y\|_2^2$ has a gradient and we want to exploit it $\Rightarrow$ Use proximal gradient!

Proposition

$\forall x \in \mathbb{R}^d$, the subdifferential of $\|\cdot\|_1$ at x is the set of vectors $\partial \|.\|_1(x) \subseteq \mathbb{R}^d$ such that

$\forall g \in \partial \|.\|_1(x)$, $\forall j \in \{1, .., d\}$,

$$g_j \begin{cases} = 1 & \text{if } x_j > 0 \\ = -1 & \text{if } x_j < 0 \\ \in [-1, 1] & \text{if } x_j = 0 \end{cases}$$

g is called a subgradient of $\|\cdot\|_1$ at x

If x has d nonzero coordinates ($\|x\|_0 = d$), then

$$\partial \|\cdot\|_1(x) = \left\{ \begin{bmatrix} \mathrm{sign}(x_1) \\ \vdots \\ \mathrm{sign}(x_d) \end{bmatrix} \right\} \qquad \mathrm{sign}(t) = \begin{cases} 1 & \text{if } t>0 \\ -1 & \text{if } t<0 \end{cases}$$

$\uparrow$

$\nabla(\|\cdot\|_1)(x)$: at this point, the gradient is well-defined

If $x = 0_{\mathbb{R}^d}$,

$$\partial \|\cdot\|_1 (0_{\mathbb{R}^d}) = \{ g \in \mathbb{R}^d \mid g_j \in [-1,1] \ \forall j = 1..d \}$$
$$= \{ g \in \mathbb{R}^d \mid \|g\|_\infty \leq 1 \}$$

Th) If $\varphi : \mathbb{R}^d \to \mathbb{R}$ is a convex function, then

$\bar{x} \in \mathbb{R}^d$ is a minimum of $\varphi \iff 0_{\mathbb{R}^d} \in \partial \varphi(\bar{x})$

Corollary: For the LASSO problem

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad \varphi(x) = \frac{1}{2m} \|Ax - b\|_2^2 + \lambda \|x\|_1,$$

$$\bar{x} \in \underset{x \in \mathbb{R}^d}{\operatorname{argmin}} \, \varphi(x) \iff -\frac{1}{m\lambda} A^T(A\bar{x} - b) \in \partial \|\cdot\|_1(\bar{x})$$

Proof sketch: φ is convex as the sum of two convex functions

hence $\bar{x} \in \underset{x \in \mathbb{R}^d}{\operatorname{argmin}} \, \varphi(x) \iff 0_{\mathbb{R}^d} \in \partial \varphi(x)$

Let $f(x) = \frac{1}{2m} \|Ax - y\|_2^2$. f C^1

$$\forall x \ldots \partial \varphi(x) = \left\{ \nabla f(x) + g \mid g \in \partial(\lambda \|.\|_1)(x) \right\}$$

$$= \left\{ \frac{1}{m} A^T(Ax-b) + \lambda g \mid g \in \partial \|.\|_1 (x) \right\}$$

$$= \left\{ \hat{g} \in \mathbb{R}^d \mid \forall j = 1..d, \right.$$

$$\left. \hat{g}_j \begin{cases} = \left[\frac{1}{m} A^T(Ax-b)\right]_j + \lambda \,\mathrm{sign}(x_j) & \text{if } x_j \neq 0 \\ \in \left[\left[\frac{1}{m} A^T(Ax-b)\right]_j - \lambda, \left[\frac{1}{m} A^T(Ax-b)\right]_j + \lambda \right] & \text{if } x_j = 0 \end{cases} \right\}$$

$$0_{\mathbb{R}^d} \in \partial \varphi(\bar{x}) \iff 0_{\mathbb{R}^d} \in \left\{ \frac{1}{2m} A^T(A\bar{x}-b) + g \mid g \in \partial(\lambda\|.\|_1)(\bar{x}) \right\}$$

$$\iff \exists\, \bar{g} \in \partial(\lambda \|.\|_1)(\bar{x}) \text{ such that}$$

$$\frac{1}{m} A^T(A\bar{x}-b) + \bar{g} = 0_{\mathbb{R}^d}$$

$$\iff \exists \bar{g} \text{ —}, \quad \bar{g} = -\frac{1}{n} A^T(A\bar{x}-b)$$

$$\iff \frac{-1}{m} A^T(A\bar{x}-b) \in \partial(\lambda \|.\|_1)(\bar{x})$$

Since $\partial(\lambda\|.\|_1)(\bar{x}) = \{ \lambda g \mid g \in \partial \|.\|_1(\bar{x}) \}$, the last inclusion is equivalent to

$$-\frac{1}{m\lambda} A^T(A\bar{x}-b) \in \partial \|.\|_1(\bar{x})$$

↳ the condition $-\frac{1}{n\lambda} A^T(A\bar{x}-b) \in \partial\|.\|_1(\bar{x})$

means that $\forall j=1,\dots,d$,

$A=[a_1 \dots a_d]$ $\quad |a_j^T(A\bar{x}-b)| \leq n\lambda$ if $\bar{x}_j=0$ (*)

$a_j \in \mathbb{R}^n$ $\quad a_j^T A(\bar{x}-b) = n\lambda\, \mathrm{sign}(\bar{x}_j)$ if $\bar{x}_j \neq 0$

As λ increases, the condition (*) is more likely to be satisfied at the solution and thus $\bar{x}$ is more likely to have zero coordinates

N.B.: These conditions are not easy to solve for arbitrary (A,b) but they can be for specific A and b, and they also serve as convergence criterion for iterative methods

③ Subgradient and proximal gradient for ℓ_1 regularized problems

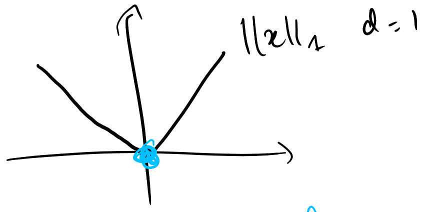
$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad f(x) + \lambda\|x\|_1 \qquad f \text{ convex} \qquad \lambda>0$$

Subgradient method

the method applies to any convex function

- Start with $x_0 \in \mathbb{R}^d$
- For $k=0,1,2,\dots$
  - Compute $g_k \in \partial(f+\lambda\|.\|_1)(x_k)$.
  - Define $x_{k+1} = x_k - \alpha_k g_k$, where $\alpha_k > 0$

↳ This method is harder to implement than a gradient-type method because: 1) the choice of subgradient matters



$x_k = 0$

$\partial(\|\cdot\|_1)(x_k) = [-1, 1]$

If we choose $g \in \partial(\|\cdot\|_1)(0)$, $g \neq 0$,

$x_k - \alpha g \neq 0$

$\Rightarrow$ we move away from the minimum

One choice that works:

$$g_k \in \operatorname{argmin}\{\|g\|_2 \mid g \in \partial h(x_k)\}$$

where h is the function to be minimized

☹ Expensive, it might require to compute the entire subdifferential

2) Sensitive to the stepsize $\alpha_k > 0$

It is possible that $g_k \in \partial h(x_k)$ and yet

$$\forall \alpha > 0, \quad h(x_k - \alpha g_k) > h(x_k)$$

↳ Nevertheless, the subgradient method and its stochastic counterpart (stochastic subgradient method for finite-sum problems) are used in training common neural architectures based on nonsmooth activations

Ex) $\mathrm{ReLU}(t) = \max(t, 0)$

Subgradient for ℓ_1 regularized problem

$$x_{k+1} = x_k - \alpha_k g_k \qquad g_k \in \partial(f + \lambda\|.\|_1)(x_k)$$

If $f \in C^1$, $\quad g_k = \nabla f(x_k) + \lambda \bar{g}_k \quad , \bar{g}_k \in \partial\|.\|_1(x_k)$

$$x_{k+1} = \underbrace{x_k - \alpha_k \nabla f(x_k)}_{\text{Gradient descent step on } f} \underbrace{- \alpha_k \lambda \bar{g}_k}_{\text{shift by } \alpha_k \lambda \bar{g}_k}$$

$\bar{g}_k \in [-1,1]^d$

→ The iterates are different from that of GD, but it is hard to see that they are sparser than the iterates of GD

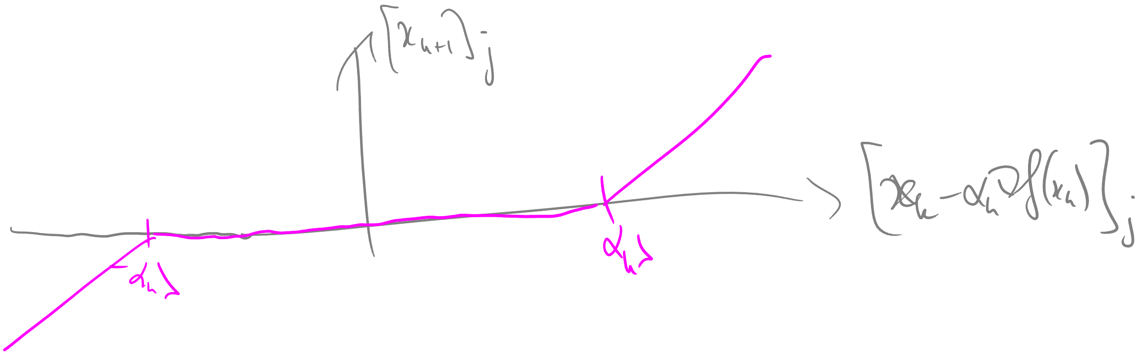
Proximal gradient iteration $(f \in C^1)$

(P) $$x_{k+1} \in \underset{x \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ f(x_k) + \nabla f(x_k)^T (x - x_k) + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 + \lambda \|x\|_1 \right\}$$

Thm The solution of the proximal subproblem (P) is unique, and defined coordinatewise by

$$\forall j = 1..d, \quad [x_{k+1}]_j = \begin{cases} [x_k - \alpha_k \nabla f(x_k)]_j - \alpha_k \lambda & \text{if } [x_k - \alpha_k \nabla f(x_k)]_j > \alpha_k \lambda \\ [x_k - \alpha_k \nabla f(x_k)]_j + \alpha_k \lambda & \text{if } [x_k - \alpha_k \nabla f(x_k)]_j < -\alpha_k \lambda \\ 0 & \text{otherwise} \end{cases}$$

This iteration sets components of x_{k+1} to 0!



- $\|x_{k+1}\|_0 \leq \|x_k-\alpha_k\nabla f(x_k)\|_0 \quad \forall k\in\mathbb{N}$

- $x_{k+1} = \mathrm{prox}_{\alpha_k\lambda\|\cdot\|_1}\left(x_k-\alpha_k\nabla f(x_k)\right)$

Remarks:
- The proximal gradient method for ℓ_1 regularization was discovered in compressed sensing under the name ISTA (Iterative Soft-Thresholding Algorithm)
- It has also been combined with acceleration (FISTA 2009)