Convex Unconstrained and Constrained Optimization

José Dorronsoro
Escuela Politécnica Superior
Universidad Autónoma de Madrid

Contents

1 Convex Optimization **3**
1.1 Convex Set and Function Basics . . . 3
1.2 Minimization of Convex Functions . . . 8
1.3 Proximal Gradient . . . 13

2 Constrained Optimization **15**
2.1 Projected Gradient . . . 15
2.2 Lagrangian Optimization . . . 17
2.3 Duality . . . 20
2.4 Support Vector Classification . . . 22

1 Convex Optimization

1.1 Convex Set and Function Basics

Learning in Machine Learning

- ML models are usually built by the minimization of a function

$$J(w) = \ell(w) + \alpha R(w),$$

  where ℓ is a loss function, R a regularizer and w varies over fixed set C

- When $C = \mathbf{R}^d$ and both ℓ and R functions are differentiable, we have to deal with an **unconstrained, differentiable** optimization problem

- When C is a proper subset of $\mathbf{R}^d$, we are dealing with a **constrained** optimization problem

- Moreover, it is often the case that either ℓ or R, or even both, are not differentiable but then they are assumed to be **convex**

Two Key Problems

- In Lasso we want to minimize

$$\begin{aligned} e(w, b) &= \frac{1}{2n} \sum_p (t^p - w \cdot x^p - b)^2 + \alpha \|w\|_1 \\ &= \frac{1}{2} mse(w, b) + \alpha \|w\|_1, \end{aligned}$$

- Here $\ell = mse$ is the mean squared error (and hence differentiable) but $R(w) = \|w\|_1 = \sum_1^d |w_i|$ is only convex but not differentiable

- In support vector classification (SVC) we want to minimize

$$\min_{w,b} f(w, b) = \frac{1}{2}\|w\|^2 + C \sum_1^n \xi^p = \frac{1}{2}\|w\|^2 + C\ell(w, b)$$

  subject to $y^p(w \cdot x^p + b) \geq 1 - \xi^p, \xi^p \geq 0$

- Here $R(w) = \|w\|_2^2$ (and hence differentiable) but the hinge loss ℓ is only convex

Optimization Scenarios

- Therefore, Lasso is an example of an **unconstrained, convex** minimization problem

- And SVC is an example of a **convex, constrained** minimization problem

- We thus need **fundamentals and techniques** to solve constrained problems with non differentiable but at least convex functions

- Moreover, convex functions are in many senses the **natural context for minimization problems**

- We will thus consider convex optimization first and then unconstrained optimization
- Reference: parts of Chapters 6, 7, 8, 9, 11 and 12 of Introduction to Nonlinear Optimization, by Amir Beck

Basic Definitions I

- We say that S is a **convex set** if for all $x, x' \in S$ and $\lambda \in [0, 1]$,

$$\lambda x + (1 - \lambda)x' \in S$$

- First we recall/clarifiy basic definitions to be more precise the kind of sets we work with
- The set $\text{int}(S) = \{x \in S : B(x, \delta) \subset S \text{ for some } \delta > 0\}$ is the **interior** of $S \subset \mathbf{R}^d$
  - If $S = \text{int}(S)$, we say that S is an **open** set
- The **closure** of S is $cl(S) = \{x : S \cap B(x, \delta) \neq \emptyset \text{ for all } \delta > 0\}$
  - If $S = cl(S)$, we say that S is a **closed** set
- **Proposition.** *S is closed iff for any sequence $\{x_n\} \subset S$ such that $x_n \to x$, then $x \in S$*
- The **boundary** of S is $\partial S = cl(S) - \text{int}(S)$

Basic Definitions II

- We say that S is **bounded** if $S \subset B(0, R)$ for some $R > 0$
- We say that S is a **compact** set if it is bounded and closed
- We state next two key results that we will use later on
- **Proposition:** *If S is a compact set, any sequence $\{x_n\} \subset S$ has a convergent subsequence $\{x_{n_k}\}$*
  - I.e., there are an $\{x_{n_k}\} \subset \{x_n\}$ and $x \in S$ such that $\lim_{k\to\infty} x_{n_k} = x$
- **Weierstrass Theorem:** *If S is a compact set and $f : S \subset \mathbf{R}^d \to \mathbf{R}$ is continuous, then f has a maximum and a minimum on S*

The Projection Theorem

- **Theorem.** *Let S be a non emtpy convex (nEC) set. Then, for any $y \notin S$, there is a unique $x \in cl(S)$ such that*

$$\|x - y\| \leq \|x' - y\| \text{ for any other } x' \in S$$

  - To prove the existence, choose any $z \in S$ and define $S_z = \{x' \in cl(S) : \|x' - y\| \leq \|z - y\|$.
  - Then S_z is closed and bounded, and since $f(z) = \|z - y\|$ is continuous, Weierstrass' theorem ensures the existence of a minimum point x
  - Uniqueness is slightly more involved but essentially elementary

- We will call the unique x the **projection** $P_S(y)$
- An important property of $P_S(y)$ is the following
- **Theorem.** *Let S be a nCE set. Then for any $y \notin S$, $x = P_S(y)$ iff $(y-x)\cdot(x'-x) \leq 0$ for all $x' \in S$*

The Supporting Hyperplane

- **Theorem.** *Let S be a nEC set and $x \in \partial S$. Then there exists a vector $p \in \mathbf{R}^d$ such that $p \cdot (x'-x) \leq 0$ for any $x' \in cl(S)$.*
  - Since $x \in \partial S$, there is a sequence $y_k \subset cl(S)^c$ such that $y_k \to x$ and, by the Projection Theorem, if $x_k = P_S(y_k)$ and $p_k = \frac{y_k - x_k}{\|y_k - x_k\|}$, then $p_k \cdot (x' - x_k) \leq 0$ for any $x' \in cl(S)$
  - Now, the sequence p_k lies in a compact subset and if $\{p_{k_j}\}$ is a convergent subsequence tends to some p, then, for any $x' \in cl(S)$,

$$p \cdot (x'-x) = \lim_j p_{k_j} \cdot (x' - x_{k_j}) \leq 0$$

- We will call the hyperplane $H = \{z : p \cdot (z-x) = 0\}$ the **supporting hyperplane**
- We can reformulate the previous theorem as saying that for a closed nEC set and $x \in \partial S$ **there is a hyperplane H that supports S at x**

Convex Functions

- Let $S \subset \mathbf{R}^d$ a nEC set; a function $f : S \to \mathbf{R}$ is **convex** if for any $x, x' \in S$ and $\lambda \in [0,1]$,

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

- f is **strictly convex** if for any $x, x' \in S$ with $x \neq x'$ and $\lambda \in (0,1)$,

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

- Convex functions have many nice properties
- **Theorem.** *Let S be a nEC set and f convex on S. Then f is continuous in int(S)*
- We define the **directional derivative** $g(x;d)$ at a point x in the direction d as the limit $\lim_{t \downarrow 0} \frac{f(x+td)-f(x)}{t}$ when it exits
  - IF f is continuously differentiable (i.e, its partials are continuous), then $g(x;d) = \nabla f(x) \cdot d$
- **Theorem.** *Let S be an nEC open set. Then $g(x;d)$ exists for any $x \in S$ and $d \in \mathbf{R}^d$*

Differentiable Convex Functions I

- **Definition.** *Let S be an nEC open set. We say that $f : S \to \mathbf{R}$ is **differentiable** at $x \in S$ if there exists a vector $\nabla f(x)$ such that for any $z \in S$*

$$f(z) = f(x) + \nabla f(x) \cdot (z-x) + \|z-x\| \alpha(x; z-x) \tag{1}$$

  such that $\lim_{z \to x} \alpha(x; z-x) = 0$

  - Equation (1) is called the **first order Taylor expansion** of f at x

- **Theorem.** *Let S be an nEC open set and f convex and continuously differentiable in S. Then, for any $x, x' \in S$ $f(x') \geq f(x) + \nabla f(x) \cdot (x' - x)$*

  - Notice that $f(\lambda x' + (1-\lambda)x) = f(x + \lambda(x' - x)) \leq \lambda f(x') + (1-\lambda)f(x)$ and hence

$$\frac{f(x + \lambda(x' - x)) - f(x)}{\lambda} \leq f(x') - f(x)$$

  and the right hand side limit when $\lambda \to 0$ is $\nabla f(x) \cdot (x' - x)$

Differentiable Convex Functions II

- For functions of a single variable the previous theorem means that **the graph of f is above its tangent at any point x**

- The previous is also sufficient and, moreover, if f is strictly convex, the inequality is strict

- **Theorem.** *Let S be an nEC open set and f differentiable in S. Then f is convex iff for any $x, x' \in S$,*

$$(\nabla f(x) - \nabla f(x')) \cdot (x - x') \geq 0$$

  - Just apply the previous theorem at x and x'

- For functions of a single variable this means that the **derivative f' is monotonously increasing**

- Because of this we will say that the gradient of a convex function is **monotone**

Differentiable Convex Functions III

- **Theorem.** *Let $f : U \in \mathbf{R}^d \to \mathbf{R}$ be twice differentiable on the open set U. Then if $B(x, r) \subset U$ and $z \in B(x, r)$, then*

$$f(z) = f(x) + \nabla f(x) \cdot (z - x) + \frac{1}{2}(z - x)^t Hf(x)(z - x) + o(\|z - x\|^2),$$

  with $Hf(x)$ the Hessian of f at x

- **Definition.** *We say that a square matrix Q is **semidefinite positive** if $w^t Q w \geq 0$ for all w. If, moreover, $w^t Q w > 0$ for all $w \neq 0$, we say that Q is **definite positive***

- We relate next convexity to the Hessians being positive definite

- **Theorem.** *Let $f : U \in \mathbf{R}^d \to \mathbf{R}$ be twice differentiable on the open convex set U. Then f is convex on U iff $Hf(x)$ is semidefinite positive for any $x \in U$. Moreover, if $Hf(x)$ is definite positive for all $x \in U$, f is strictly convex*

  - Notice that $f(x) = x^4$ is strictly convex, but $f''(x) = 12x^2$ and $f''(0) = 0$

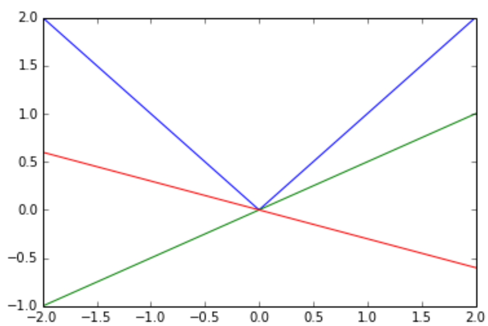
Subgradients and Subdifferentials

- Recall that our goal here are **non-differentiable** convex functions

- In fact, much of the above extends to this case if we look at gradients in an appropriate way
- **Definition.** *Let $f : S \to \mathbf{R}$ with S an nEC open set. We say that $\xi \in \mathbf{R}^d$ is a* ***subgradient*** *at $x \in S$ if for any $x' \in S$, $f(x') \geq f(x) + \xi \cdot (x' - x)$*
- **Definition.** *The subset $\partial f(x) = \{\xi : \xi$ is a subgradient of f at $x\}$ is called the* ***subdifferential*** *of such an f at x*
- Our next goal is to show that for such an f and $x \in S$, $\partial f(x) \neq \emptyset$
- **Definition.** *Let $f : S \to \mathbf{R}$ with S an nEC open set. The* ***epigraph*** *of f is the set $epi(f) = \{(x, t) : x \in S, t \geq f(x)\}$*
- **Proposition.** *Let $f : S \to \mathbf{R}$ with S an nEC open set. Then f is convex iff $epi(f)$ is convex*

An Example

- Consider $f(x) = |x|$



- It is convex and differentiable in all $\mathbf{R}$ but 0
- At 0 we have $\partial f(0) = [-1, 1]$
- Its epigraph is obviously convex

Existence of Subgradients and Subdifferentials

- **Theorem.** *Let $f : S \to \mathbf{R}$ be a convex function on the nEC open set S. Then, for all $x \in int(S)$, $\partial f(x) \neq \emptyset$*
  - Since $(x, f(x)) \in \partial\text{epi}(f)$, there is a hyperplane $f(x) + \xi(x' - x)$ that supports epi(f) at $(x, f(x))$. But then $\xi \in \partial f(x)$
- This has a converse result
- **Theorem.** *Let $f : S \to \mathbf{R}$ with S an nEC open set. Then, if for all $x \in int(S)$, $\partial f(x) \neq \emptyset$, f is a convex function*

- Things are much simpler for differentiable functions
- **Theorem.** *Let $f : S \to \mathbf{R}$ be a convex function on the nEC open set S. If f is differentiable at $x \in int(S)$, then $\partial f(x) = \{\nabla f(x)\}$*

Moreau-Rockafellar Theorem

- **Theorem.** *Let $f, g : S \to \mathbf{R}$, with S an nEC open set, be two convex functions. Then, as subsets,*
$$\partial f(x) + \partial g(x) = \partial(f+g)(x)$$
for any $x \in S$
- Often one allows convex functions to take a $+\infty$ value, although never $-\infty$; in this case there is a more general version of Moreau-Rockafellar
- In this case we can consider any such function initially defined on a subset S as defined on the entire $\mathbf{R}^d$ by setting $f(x) = +\infty$ for $x \notin S$
- For such an f we define $\text{dom}(f) = \{x \in \mathbf{R}^d : f(x) < +\infty\}$
- **Theorem.** *Let $f, g : \mathbf{R}^d \to (-\infty, +\infty]$ be two convex functions. Then, as subsets, $\partial f(x) + \partial g(x) \subset \partial(f+g)(x)$ for any $x \in S$ Moreover, if $int(dom(f)) \cap int(dom(g)) \neq$, $\partial(f+g)(x) \subset \partial f(x) + \partial g(x)$*

1.2 Minimization of Convex Functions

Minima of Convex Functions

- Convex functions may not have a minimum (think of $f(x) = x$) but when they do, they have nice properties
- Let S be a nEC set, $f : S \to \mathbf{R}$ a convex differentiable function and consider the following problem:
$$\min_{x \in S} f(x) \tag{2}$$
- **Theorem.** *Assume $x^* \in S$ is a local solution of (2). Then x^* is also a global minimum of (2) Moreover, if f is strictly convex, x^* is the unique global minimum*
  - We know that for some $\delta > 0$, $f(x) \leq f(z)$ for all $z \in B(x, \delta)$
  - Now if $x' \in S$ verifies $f(x') < f(x)$ and λ is small enough, we can get $z = \lambda x' + (1 - \lambda)x \in B(x, \delta) \cap S$, but then
  $$f(z) \leq \lambda f(x') + (1-\lambda) f(x) < f(x)$$
  .

Minima and Subgradients

- **Theorem.** *Let S be a nEC set and $f : S \to \mathbf{R}$ a convex function. Then, $x^* \in S$ solves (2) iff there is a $\xi \in \partial f(x^*)$ such that $\xi \cdot (x - x^*) \geq 0$ for any other $x \in S$*

  - The sufficiency is essentially obvious: since f is convex and $\xi \in \partial f(x^*)$, we have for any other $x \in S$,
$$f(x) \geq f(x^*) + \xi \cdot (x - x^*) \geq f(x^*)$$
  - Necessity is harder as we have to deal with a general convex S and the minimum x^* may be in its boundary ∂S

- The preceding result simplifies for differentiable functions
- **Theorem.** *Let S be a nEC set and $f : S \to \mathbf{R}$ a convex differentiable function. Then, $x^* \in S$ solves (2) iff $\nabla f(x^*) \cdot (x - x^*) \geq 0$ for any other $x \in S$*

Fermat's Theorem

- **Fermat's Theorem.** *Let S be an open nEC set and $f : S \to \mathbf{R}$ a convex function. Then, $x^* \in S$ solves (2) iff $0 \in \partial f(x^*)$*
  - The sufficiency is again obvious.
  - So is here the necessity: if x^* is a global minimum, for any $x \in S$, $f(x) \geq f(x^*) = f(x^*) + 0 \cdot (x - x^*)$ and, thus, $0 \in \partial f(x^*)$
- Again the preceding simplifies for differentiable functions
- **Theorem.** *Let S be an open nEC set and $f : S \to \mathbf{R}$ a convex differentiable function. Then, $x^* \in S$ solves (2) iff $\nabla f(x^*) = 0$*

Examples

- Consider again $f(x) = |x|$;
  - It has a minimum at 0 and $0 \in \partial f(0)$
- A second example is the hinge loss $h(x) = \max\{-x, 0\}$, with minima in the set $M = [0, \infty)$
  - Here $0 \in \partial h(0) = [-1, 0]$ and $\partial h(x) = \{0\}$ if $x > 0$
- A third example are the ReLU activations $r(x) = \max\{0, x\}$ used in DNNs
  - By the way, DNNs do not bother much with differentiability niceties

Towards the Proximal Operator

- The preceding shows that convex functions are the **natural ones** to study function minimization
- In fact, one can aim to derive **general algorithms** to find their minima, in contrast to the situation for general functions
- The tool to achieve this is the **proximal operator**
- If a convex f has a minimum at x, we have $0 \in \lambda \partial f(x)$ for all $\lambda > 0$ and, thus,
$$0 \in \partial \lambda f(x) \text{ iff } x \in x + \lambda \partial f(x) = (I + \lambda \partial f)(x) \tag{3}$$

- Thus, if we could invert $I + \lambda\partial f$, the minimum will verify $x = (I + \lambda\partial f)^{-1}(x)$

Back to $|x|$

- The minimum of $|x|$ is 0 and we have $0 \in \partial|\cdot|(0)$
- We have

$$\begin{aligned}(I + \lambda\partial|\cdot|)(x) &= x - \lambda \text{ if } x < 0\\ &= [-\lambda, \lambda] \text{ if } x = 0\\ &= x + \lambda \text{ if } x > 0\end{aligned}$$

- Although not a function, $I + \lambda\partial|\cdot|$ is increasing, and we can invert it by flipping it around the $y = x$ line, to get

$$\begin{aligned}(I + \lambda\partial|\cdot|)^{-1}(y) &= y + \lambda \text{ if } y < -\lambda\\ &= 0 \text{ if } y = 0\\ &= y - \lambda \text{ if } y < \lambda\end{aligned}$$

- Or just $(I + \lambda\partial|\cdot|)^{-1}(y) = \text{sign}\,(y)[|y| - \lambda]_+ = \text{soft}_\lambda(y)$

Monotone Operators

- We could invert $I + \lambda\partial|\cdot|$ because it is essentially a monotone function
- The set-valued operator $T : R^d \to 2^{R^d}$ is called **monotone** if for all x_1, x_2, $\xi_1 \in T(x_1)$, $\xi_2 \in T(x_2)$ we have $(\xi_1 - \xi_2) \cdot (x_1 - x_2) \geq 0$.
- **Theorem.** *If f is a convex function, ∂f is a monotone operator*
  - This follows from the subgradient's definition: take x_1, x_2, $\xi_1 \in T(x_1)$, $\xi_2 \in T(x_2)$; then

$$\begin{aligned}f(x_2) &\geq f(x_1) + \xi_1 \cdot (x_2 - x_1)\\ f(x_1) &\geq f(x_2) + \xi_2 \cdot (x_1 - x_2) = f(x_2) - \xi_2 \cdot (x_2 - x_1)\end{aligned}$$

    and just add these two inequalities

Inverting $I + \lambda\partial f$

- While in principle, $(I + \lambda\partial f)^{-1}$ is defined as a set function:

$$(I + \lambda\partial f)^{-1}(x) = \{z : x \in (I + \lambda\partial f)(z)\},$$

  it is actually a standard point function

- **Theorem.** *The set function $(I + \lambda\partial f)^{-1}$ is a single valued function*
  - This follows from the monotonicity of ∂f

– If $(I + \lambda\partial f)^{-1}$ is not single valued, there are two $z, z' \in (I + \lambda\partial f)^{-1}(x)$, that is, there are $\xi, \xi' \in \partial f(x)$ such that

$$\begin{aligned} x = z + \lambda\xi = z' + \lambda\xi' \quad &\Rightarrow \quad z - z' + \lambda(\xi - \xi') = 0 \\ &\Rightarrow \quad z - z' = -\lambda(\xi - \xi') \end{aligned}$$

– But since ∂f is monotone, we arrive at $z = z'$, as we have

$$0 \leq (z - z') \cdot (\xi - \xi') = -\frac{\|z - z'\|^2}{\lambda}$$

Understanding the Proximal Operator

- We call $(I + \partial f)^{-1}(x)$ the **proximal operator** prox_f
- An equivalent and slightly more practical definition is
- **Proposition.** *We have*

$$\text{prox}_f(x) = \arg\min_u \left\{ f(u) + \frac{1}{2}\|u - x\|^2 \right\} \qquad (4)$$

  – We have that p is the minimum of (4) iff $0 \in p - x + \partial f(p)$ iff $x \in (I + \partial f)(p)$ iff $p = (I + \partial f)^{-1}(x)$

- For a C^1 function f, $p = p_\lambda(x) = \text{prox}_{\lambda f}(x)$ solves the equation

$$\lambda\nabla f(p) + p - x = 0, \text{ that is, } p = x - \lambda\nabla f(p)$$

- Thus, in this case, the proximal corresponds to an **implicit** gradient descent with step λ

Fixed Points

- The following theorem re-states much of the preceding
- **Theorem.** *Let S be an open nEC set and $f : S \to \mathbf{R}$ a convex function. Then, $x^* \in S$ solves* (2) *iff x is a fixed point of $(I + \partial\lambda f)^{-1}$*
- This suggests to try to obtain fixed points of an operator T is to start from some x_0 and study the convergence of the iterations $x_{k+1} = T(x_k)$
- We say that the operator T is **contractive** if there is a $\lambda < 1$ such that for all x, x', $\|T(x) - T(x')\| \leq \lambda\|x - x'\|$
- In other words, T is Lipschitz with a constant $\lambda < 1$

Picard's Theorem

- **Picard's Theorem.** *If T is a contractive operator, the sequence $x_{k+1} = T(x_k)$ converges to the unique fixed point of T*

  – The key is that contractivity implies that x_k is a Cauchy sequence

- First is easy to see that x_n is bounded, i.e., $\|x_n\| \leq R$ for some R
- For any pair n, k consider $\|x_{n+k} - x_n\|$; we have

$$\begin{aligned} \|x_{n+k} - x_n\| &\leq \lambda \|x_{n-1+k} - x_{n-1}\| \leq \lambda^2 \|x_{n-2+k} - x_{n-2}\| \leq \dots \\ &\leq \lambda^n \|x_k - x_0\| \leq 2\lambda^n R \end{aligned}$$

  and now is easy to check the Cauchy's sequence definition
- x_n has thus a limit x^* but then $\lim T(x_n) = \lim x_{n+1} = x^*$

Non Expansive Operators

- Unfortunately, prox_f is not contractive
  - It it were so, it would have a unique fixed point, i.e., f would have a unique minimum
  - In fact, if f is strictly convex, prox_f is contractive
- In general the proximal operator satisfies a milder condition
- **Definition.** *An operator T is is* **firmly non expansive** *if*

$$\|T(x_1) - T(x_2)\|^2 \leq (x_1 - x_2) \cdot (T(x_1) - T(x_2))$$

  - It follows from this that $\|T(x_1) - T(x_2)\| \leq \|x_1 - x_2\|$, i.e., T is Lipschitz with constant 1
- **Proposition.** *The proximal operator is firmly non expansive*
- We cannot use Picard's theorem to arrive at a fixed point but we still have
- **Proposition.** *If the convex f has a minimum, the sequence $\text{prox}_{\lambda f}(x_k)$ converges to a minimizer of f*

The Proximal Algorithm

- **Theorem.** *Let the convex function $f : \boldsymbol{R}^d \to \mathbf{R}$ have a minimum. Then, for any sequence λ_k such that $\sum_k \lambda_k = \infty$, the sequence*

$$x_{k+1} = (I + \lambda_k \partial f)^{-1}(x_k)$$

  converges to a minimizer x^* of f
- Thus, if for a convex f we can compute its proximal, we have a **general algorithm to find a minimizer**
- However, computing prox_f for a general convex f is quite **difficult and/or costly**

Computing the Proximal Operator

- In some cases the definition $(I + \partial f)^{-1}(x)$ makes it easy to compute the prox_f operator
- Also, when $f(x, y)$ separates as $f(x, y) = g(x) + h(y)$, its proximal also separates as

$$\text{prox}_f(u, v) = (\text{prox}_g(u), \text{prox}_h(v)) \tag{5}$$

- In general, equation (4) allows the computation of the proximal operator as a minimization problem
- But although amenable to an algorithmic resolution, it is in general still quite a difficult problem

Takeaways on Convex Minimization

- Convex functions only have global minima (if they do)
- Even when non differentiable, they have subgradients
- A convex f has a minimum at x^* on an open convex set iff $0 \in \partial f(x^*)$ and, moreover, iff

$$x^* = (I + \lambda\partial f)^{-1}(x^*) = \text{prox}_{\lambda f}(x^*)$$

- Thus, we can in principle minimize convex functions by finding iteratively fixed points of prox
- In fact, the sequence obtained iterating from an initial point converges to a minimum x^*
- Bu this is practical provided prox can be computed without much work ... which is often not the case

1.3 Proximal Gradient

Minimizing Sums of Convex Functions

- A frequent situation is to solve for f, g both convex with f also C^1 (i.e., continuous with continuous partials), problems of the form

$$\min_{x \in \mathbf{R}^d} F(x) = f(x) + g(x) \tag{6}$$

- We know that x^* solves (6) iff for any $\lambda > 0$

$$0 \in \lambda\partial(f + g)(x^*) = \lambda\nabla f(x^*) + \lambda\partial g(x^*)$$

  or, in other words, there is a $\xi \in \partial g(x^*)$ s.t. $0 = \lambda\nabla f(x^*) + \lambda\xi$

- But then we have $0 = \lambda\nabla f(x^*) - x^* + x^* + \lambda\xi \in \lambda\nabla f(x^*) - x^* + (I + \lambda\partial g)(x^*)$, i.e.

$$x^* - \lambda\nabla f(x^*) \in (I + \lambda\partial g)(x^*)$$

- Or, equivalently

$$x^* = (I + \lambda\partial g)^{-1}(x^* - \lambda\nabla f(x^*)) = \text{prox}_{\lambda g}(x^* - \lambda\nabla f(x^*))$$

The Proximal Gradient Method

- This leads to the **Proximal Gradient Method** with iterations of the form

$$x_{k+1} = \text{prox}_{\lambda_k g}(x_k - \lambda_k\nabla f(x_k)) \tag{7}$$

- **Theorem.** *Assume that ∇f is Lipschitz with constant L. Then, for any $\lambda < \frac{1}{L}$, the iterations* (7) *with $\lambda_k = \lambda$ verify $F(x_k) \to F^*$, with F^* the minimum of* (6) *and, moreover*

$$F(x_k) - F^* = O\left(\frac{1}{k}\right)$$

- Notice that for $g = 0$, (7) reduces to gradient descent, and for $f = 0$ to proximal minimization
- The **Lasso problem** is a particular case of the above

The Lasso Problem

- Recall that in Lasso we want to minimize

$$\begin{aligned} e(w, b) &= \frac{1}{2n} \sum_p (t^p - w \cdot x^p - b)^2 + \alpha \|w\|_1 \\ &= \frac{1}{2} mse(w, b) + \alpha \|w\|_1, \end{aligned}$$

with mse C^1 and $\|\cdot\|_1$ convex but not differentiable at 0

  - We will assume x, y to be centered to ensure $b = 0$ and then work just with $e(w)$

- Then, w^* is optimal for e iff $0 \in \frac{\lambda}{2} \nabla mse(w^*) + \lambda\alpha\partial\|\cdot\|_1(w^*)$ for all $\lambda > 0$ or, equivalently,

$$w^* - \frac{\lambda}{2} \nabla mse(w^*) \in (I + \lambda\alpha\partial\|\cdot\|_1)(w^*)$$

- That is, $w^* = (I + \lambda\alpha\partial\|\cdot\|_1)^{-1}(w^* - \frac{\lambda}{2}\nabla mse(w^*))$

Solving Lasso

- Now, if X is the $n \times d$ sample matrix and Y the $n \times 1$ target vector, we have

$$\begin{aligned} mse(w) &= \frac{1}{n} \|Xw - Y\|^2 = \frac{1}{n} (w^t X^t - Y^t)(Xw - Y) \\ &= \frac{1}{n} (w^t X^t X w - 2 w^t X^t Y + Y^t Y) \end{aligned}$$

- The gradient is thus

$$G = \nabla mse(w) = \frac{2}{n} (X^t X w - X^t Y) = \frac{2}{n} X^t (Xw - Y),$$

and, componentwise, $G_j = \frac{2}{n} \sum_1^n x_j^p (x^p \cdot w - y^p), \ \ 1 \le j \le d$

- Now $\|w\|_1 = \sum_1^d |w_i|$ separates as a sum of single valued functions and by (5),

$$\left[\mathrm{prox}_{\lambda\|\cdot\|}(z)\right]_i = \mathrm{sign}(z_i)\, [|z_i| - \lambda]_+ = \mathrm{soft}_\lambda(z_i), \ \ 1 \le i \le d$$

Proximal Gradient for Lasso

- Putting all this together, we have

$$w_{k+1} \quad = \quad \mathbf{soft}_{\lambda\alpha}\left(w^k - \frac{\lambda}{n}X^t(Xw^k - Y)\right)$$

  with $\mathbf{soft}_\mu(z)_i = soft_\mu(z_i)$

- This is known as the ISTA algorithm and has a convergence rate of $O(1/k)$
- If known, one chooses $\lambda = \frac{1}{L}$, with L the Lipschitz constant of $\nabla mse(w)$
- However, for the Lasso specific case, the GLMNet algorithm is more efficient

Lasso Variants

- Lasso's advantage: thresholding forces non relevant coeffiicents to zero
  - It can be used for feature selection
- However, Lasso models often underperform ridge regression
- Solution: **Elastic Nets**, which minimizes

$$e_{EN}(w) = mse(w) + \frac{\alpha_2}{2}\|w\|^2 + \alpha_1\|w\|_1$$

- ISTA's iteration is now

$$w_{k+1} = \text{soft}_{\frac{\alpha_1}{L}}\left(w_k - \frac{1}{L}\left(\nabla mse(w_k) + \alpha_2 w_k\right)\right)$$

- Other, related algorithms are **group Lasso, fused Lasso**, as well as logistic regression variants for classification
- They are all **linear models**
  - Also often used for feature selection
  - But weaker than MLPs or SVMs

2 Constrained Optimization

2.1 Projected Gradient

Projected Gradient

- For f a C^1 function and a closed nEC S, consider the problem

$$\min_{x\in C} f(x) \tag{8}$$

- Defining $\imath(x) = 0$ if $x \in C$ and $+\infty$ if $x \notin C$, we can write (8) as

$$\min_{x\in\mathbf{R}^d} f(x) + \imath(x) \tag{9}$$

- Thus, if f is convex, x^* solves (9) iff for all $\lambda > 0$
$$x^* = \text{prox}_{\lambda\iota_C}\left(x^* - \lambda\nabla f(x^*)\right)$$
- We need to compute prox_{ι_C}
- **Proposition.** *We have* $\text{prox}_{\lambda\iota_C}(x) = P_C(x)$
  - Just use the characterization (4) of the proximal operator
- **Proposition.** *If f is convex, x^* solves* (9) *iff*
$$x^* = P_C(x^* - \lambda\nabla f(x^*))$$

Projected Gradient

- The previous results lead us to the **Projected Gradient** algorithm to solve (8)

Algorithm 1: Projected Gradient

```
function projected_gradient(ε, x_0) is
    k = 0
    for k = 1, 2, ... do
        choose a step λ_k
        x_{k+1} = P_C(x_k − λ_k ∇f(x_k))
        if ‖x_{k+1} − x_k‖ ≤ ε then
            return x_{k+1}
        end
    end
end
```

- It has the convergence properties of the Proximal Gradient algorithm

Have We Finished?

- Yes if we could compute projections over general convex sets
  - But this is easy only for particular sets
- If $C = B(x, \delta)$ and $z \notin B(x, \delta)$, $P_C(z) = x + \delta\frac{z-x}{\|z-x\|}$
  - This is relevant for the constrained formulation of Ridge regression
$$\min_{w,b} mse(w, b) \text{ s.t. } \|w\|_2 \le \rho$$
- If C is the positive orthant $C = \{x : x_i \ge 0,\ 1 \le i \le d\}$, $P_C(x)_i = \max\{0, x_i\}$
  - This is relevant for homogeneous support vector classification
- But it is much harder to compute the projection on the 1-norm ball
  - This is needed for constrained Lasso
$$\min_{w,b} mse(w, b) \text{ s.t. } \|w\|_1 \le \rho$$
  - Trying to solve Lasso this way won't be easier than by ISTA
  - The same is true for P_C on a general convex C and we need new ideas to solve (8) in practice

2.2 Lagrangian Optimization

Basics of Lagrange Multipliers

- For $f, g : \mathbf{R}^2 \to \mathbf{R}$ consider the following minimization problem

$$\min f(x, y) \text{ s. t. } h(x, y) = 0 \tag{10}$$

- Assuming the **implicit function theorem** holds, we can find a function $y = \phi(x)$ s.t. $h(x, \phi(x)) = 0$ and, thus, we can write

$$f(x, y) = f(x, \phi(x)) = \Psi(x)$$

- At a minimum x^* with $y^* = \phi(x^*)$ we thus have

$$0 = \Psi'(x^*) = \frac{\partial f}{\partial x}(x^*, y^*) + \frac{\partial f}{\partial y}(x^*, y^*)\phi'(x^*) \tag{11}$$

- But since $h(x, \phi(x)) = 0$, we also have

$$0 = \frac{\partial h}{\partial x}(x^*, y^*) + \frac{\partial h}{\partial y}(x^*, y^*)\phi'(x^*) \Rightarrow \phi'(x^*) = -\frac{\frac{\partial h}{\partial x}(x^*, y^*)}{\frac{\partial h}{\partial y}(x^*, y^*)} \tag{12}$$

Basics of Lagrange Multipliers II

- Putting together (11) and (12) we arrive at

$$0 = \frac{\partial f}{\partial x}(x^*, y^*)\frac{\partial h}{\partial y}(x^*, y^*) - \frac{\partial f}{\partial y}(x^*, y^*)\frac{\partial h}{\partial x}(x^*, y^*)$$

- That is, at (x^*, y^*), $\nabla f \perp \left(\frac{\partial h}{\partial y}, -\frac{\partial h}{\partial x}\right)$ and, since $\left(\frac{\partial h}{\partial y}, -\frac{\partial h}{\partial x}\right) \perp \nabla h$, we have $\nabla f \parallel \nabla h$, i.e. $\nabla f(x^*, y^*) = -\mu^* \nabla h(x^*, y^*)$ for some $\lambda^* \neq 0$
- Thus, for the **Lagrangian**

$$L(x, y, \lambda) = f(x, y) + \mu h(x, y),$$

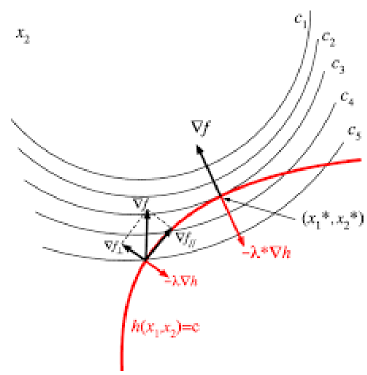
  we have that at a minimum (x^*, y^*) there is a $\mu^* \neq 0$ s.t.

$$\nabla L(x^*, y^*, \lambda^*) = \nabla f(x^*, y^*) + \mu^* \nabla h(x^*, y^*) = 0 \tag{13}$$

- Thus a way to solve (10) is to define its Lagrangian and solve simultaneously (13) and the constraint $h(x, y) = 0$

Basics of Lagrange Multipliers III

- Graphically we have
- We consider next how these ideas are applied in a general context



Inequality Constrained Minimization

- Consider the following minimization problem
$$\min f(x) \text{ s. t. } g_i(x) \leq 0,\ i = 1, \ldots, m \tag{14}$$
with f and the g_i being C^1 functions
- An x that verifies the constraints is said to be **feasible**
- A feasible x^* is a **local minimum** of (14) if there is a $\delta > 0$ s.t. $f(x^*) \leq f(x)$ for all $x \in B(x^*, \delta) \cap \{g_i \leq 0, i = 1, \ldots, m\}$
- **Proposition.** *Assume x^* is a local minimum of* (14) *and let $A(x^*) = \{i : g_i(x^*) = 0\}$ be the set of **active constraints**. Then, there is no $d \in \mathbf{R}^d$ s.t. $\nabla f(x^*) \cdot d < 0$ and $\nabla g_i(x^*) \cdot d < 0$ for all $i \in A(x^*)$*
  - If such a **descent direction** d exists, we will have for t small, $f(x^* + td) < f(x^*)$, $g_i(x^* + td) < g_i(x^*) \leq 0$; hence, x^* won't be a minimum

The Fritz John Conditions

- **Theorem. Fritz John's Conditions.** *Let x^* be a local minimum of* (14). *There is a λ_0 and $\lambda_i \geq 0$, $1 \leq i \leq m$, not all 0, s.t.*
$$\begin{aligned} \lambda_0 \nabla f(x^*) + \sum_1^m \lambda_i \nabla g_i(x^*) &= 0, \\ \lambda_i g_i(x^*) &= 0 \end{aligned} \tag{15}$$
  - Notice that if $i \notin A(x^*)$, $g_i(x^*) < 0$ and, hence, $\lambda_i = 0$
- It may be the case that, in the above, $\lambda_0 = 0$, which then implies that the $\nabla g_i(x^*)$ would be linearly dependent
  - But this may very well happen when x^* is not a local minimimum

  - And we wouldn't get any information about f

- Thus, to exploit the above conditions to locate a global minimum, we must enforce $\lambda_0 \neq 0$
- The simplest way is to ensure this is that the $\nabla g_i(x^*)$ are linearly independent

The KKT Conditions

- **Theorem. KKT Conditions.** *Let x^* be a local minimum of* (14) *and assume that $\{\nabla g_i(x^*) : i \in A(x^*)\}$ are linearly independent. Then, there are $\lambda_i \geq 0$, $1 \leq i \leq m$, not all 0 s.t.*

$$\begin{aligned} \nabla f(x^*) + \sum_1^m \lambda_i \nabla g_i(x^*) &= 0, \\ \lambda_i g_i(x^*) &= 0 \end{aligned}$$

  - Just notice that the Fritz John conditions (15) must hold, but if $\lambda_0 = 0$, the $\{\nabla g_i(x^*)\}$ would be linearly dependent.
  - Thus, we must have $\lambda_0 \neq 0$ and we just have to divide by λ_0 to arrive at (16)

General Constrained Minimization

- Consider the following minimization problem

$$\min f(x) \text{ s. t. } \begin{aligned} g_i(x) &\leq 0,\ i = 1, \ldots, m \\ h_i(x) &= 0,\ i = 1, \ldots, p \end{aligned} \qquad (16)$$

  with f and the g_i, h_j being C^1 functions

- **Theorem. KKT Conditions.** *Let x^* be a local minimum of* (16) *and assume that*

$$\{\nabla g_i(x^*) : i \in A(x^*)\} \cup \{\nabla h_j(x^*) : 1 \leq j \leq m\}$$

  are linearly independent. Then, there are $\lambda_i \geq 0$, $1 \leq i \leq m$, not all 0, and $\mu_1, \ldots, \mu_p$ s.t.

$$\begin{aligned} \nabla f(x^*) + \sum_1^m \lambda_i \nabla g_i(x^*) + \sum_1^p \mu_j \nabla h_j(x^*) &= 0, \\ \lambda_i g_i(x^*) &= 0 \end{aligned} \qquad (17)$$

Regular and KKT Points

- To lighten the statements we define regular and KKT points
- We say that a feasible point x is **regular** if

$$\{\nabla g_i(x) : i \in A(x)\} \cup \{\nabla h_j(x) : 1 \leq j \leq m\}$$

  are linearly independent

- We say that a feasible point x is a **KKT point** if conditions (17) hold at x

- We can thus rewrite the previous theorem as stating that, under its conditions, **if a minimum point x^* is regular, then it is a KKT point**
- The KKT conditions give us a set of equations that a minimum must verify
  - We can try to solve them and check then that the solution is indeed a minimum

The Convex Case

- Until now we have just seen **necessary** conditions; in the convex case they are also **sufficient**
- **Theorem.** *If in Problem* (16) *we assume f and the g_i to be convex and the h_j to be affine, then a regular KKT point x^* is an optimum of Problem* (16)
- This is the situation in several key problems in ML
  - The constrained versions of Ridge and Lasso, with Lagrangians

$$
\begin{aligned}
L(w, \lambda) &= \frac{1}{2} mse(w) + \frac{\lambda}{2}(\|w\|_2^2 - \rho) \\
L(w, \lambda) &= \frac{1}{2} mse(w) + \lambda(\|w\|_1 - \rho)
\end{aligned}
$$

    * Notice that dropping the ρ term we get their standard unconstrained versions
  - The primal and dual versions of **support vector classification and regression**

The Slater Conditions

- Checking the regularity of a given point may be hard in general
- Slater's conditions simplify this for the convex case
- We say that a point x verifies the **Slater conditions** for problems (14) and (16) if $g_i(x) < 0$ for all $i = 1, \ldots, m$
- **Theorem.** *Let x^* be a solution for Problem* (14) *with f and g_i being C^1 and the g_i also convex and assume the problem has a Slater point. Then x^* is a KKT point*
- **Theorem.** *Let x^* be a solution for Problem* (16) *where we assume the f and g_i C^1 to be C^1 and convex, and the h_j affine. Then, if the problem has a Slater point, a KKT point x^* is also an optimum*

2.3 Duality

The Lagrangian and the Dual Problem

- Consider the following minimization problem

$$
\min f(x) \text{ s. t. } \quad
\begin{aligned}
g_i(x) &\leq 0, \ i = 1, \ldots, m \\
h_i(x) &= 0, \ i = 1, \ldots, p
\end{aligned}
$$

- We define for $\lambda_i \geq 0$, μ_j, $1 \leq i \leq m$, $1 \leq j \leq p$, the **Lagrangian** as

$$L(x, \lambda, \mu) = f(x) + \sum_i \lambda_i g_i(x) + \sum_j \mu_j h_j(x)$$

  - Notice that at a feasible x we have $L(x, \lambda, \mu) \leq f(x)$

- We define the **dual function** with domain dom $(q) = \mathbf{R}_+^m \times \mathbf{R}^p$ as

$$q(\lambda, \mu) = \min_x L(x, \lambda, \mu)$$

- Then the **dual problem** is

$$\max_{(\lambda,\mu)\in \text{dom } (q)} q(\lambda, \mu) \tag{18}$$

Weak Duality

- **Proposition.** *dom(q) is a convex set and q a concave function*
  - Hence, $-q$ is convex
- **Theorem. Weak Duality** *If f^* and q^* are optimal values for problems* (16) *and* (18)*, respectively, then $q^* \leq f^*$*
  - Notice that for any $(\lambda, \mu) \in$ dom (q)

$$\begin{aligned} q(\lambda, \mu) &= \min_x L(x, \lambda, \mu) \leq \min_{x \text{ feasible}} L(x, \lambda, \mu) \\ &\leq \min_{x \text{ feasible}} f(x) = f^* \end{aligned}$$

  and, hence, $q^* \leq f^*$

Strong Duality

- In general, there is no guarantee that $f^* = q^*$; however, this is so in the convex case
- **Theorem. Strong Duality** *Consider problem* (14) *where f and the g_i are C^1 and convex and there is a Slater point. Then, if f^* is the optimal value of* (14)*,* (18) *has an optimal value q^* and $q^* = f^*$*
- **Theorem. Strong Duality II** *Consider problem* (16) *where f and the g_i are C^1 and convex, and the h_j are affine. Then, if there is a Slater point and f^* is the optimal value of* (16)*,* (18) *has an optimal value q^* and $q^* = f^*$*

And So What?

- Notice that the dual constraints will be in most cases much simpler that the primal ones
- If we can compute the dual function and strong duality holds, it will be worth our while to
  - Try first to solve the dual problem (18) to get optimum λ^*, μ^*
  - Try then to get a primal optimum solution x^* and its value f^* from the dual solution
- Usually, once we have got the dual solutions λ^*, μ^*, we may try to **exploit the KKT conditions** to derive from them a primal solution x^*
- This is precisely the approach followed for **support vector machines**

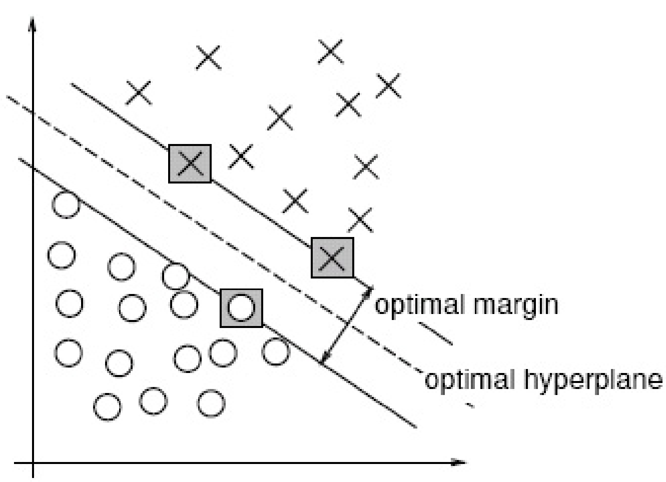
2.4 Support Vector Classification

Revisiting the Classification Problem

- Basic problem: binary classification of a sample
$$S = \{(x^p, y^p), 1 \le p \le N\}$$
with d–dimensional x^p patterns and $y^p = \pm 1$
- We assume that S is **linearly separable**: for some w, b
$$\begin{aligned} w \cdot x^p + b &> 0 \quad \text{if } y^p = 1; \\ w \cdot x^p + b &< 0 \quad \text{if } y^p = -1 \end{aligned}$$
- More concisely, we want $y^p(w \cdot x^p + b) > 0$
- How can we find a pair w, b so that the model **generalizes well**?

Margins and Generalization

- Intuitively, we will have good generalization if (w, b) has a large **margin**



- But, how can we **ensure a maximum margin?**

Distance to a Hyperplane

- Recall that given the hyperplane $\pi : w \cdot x + b = 0$, w is orthogonal to the surface defined by π
- If $x_0 \in \pi$, we compute the distance $d(x, \pi)$ of a point x to π projecting on w the vector $\overrightarrow{x_0 x}$, i.e.
$$d(x, \pi) = \frac{|w \cdot \overrightarrow{x_0 x}|}{\|w\|} = \frac{|w \cdot x - w \cdot x_0|}{\|w\|} = \frac{|w \cdot x + b|}{\|w\|}$$
for $w \cdot x_0 + b = 0$; i.e. $w \cdot x_0 = -b$

- The absolute values compensate for the orientation of w
- When the origin is in π (homogeneous π), the distance is

$$d(x, \pi) = \frac{|w \cdot x|}{\|w\|}$$

Learning and Margins

- If we assume w "points" to the positive patterns, we have $y^p(w \cdot x^p + b) = |w \cdot x^p + b|$
- The **margin** $\gamma = \gamma(w)$ is precisely the **minimum distance** between the sample S and π, i.e.,

$$\gamma = m(w, b, S) = \min_p d(x^p, \pi) = \min_p \frac{y^p(w \cdot x^p + b)}{\|w\|}$$

- Notice that $(\lambda w, \lambda b)$ give the same margin than (w, b); we can thus normalize (w, b) as we see fit
- For instance, taking $\|w\| = 1$ we have

$$\gamma(w) = \min_p \frac{y^p(w \cdot x^p + b)}{\|w\|} = \min_p y^p(w \cdot x^p + b)$$

Hard Margin SVC

- But we will work with the following normalization of w, b

$$\min_p y^p(w \cdot x^p + b) = 1$$

  - Since S is finite, we will have $y^{p_0}(w \cdot x^{p_0} + b) = 1$ for some p_0
- For a pair w, b so normalized we then have

$$m(w, b) = \min_p \left\{ \frac{y^p(w \cdot x^p + b)}{\|w\|} \right\} = \frac{y^{p_0}(w \cdot x^{p_0} + b)}{\|w\|} = \frac{1}{\|w\|}$$

- Thus, we work with these w and maximize $1/\|w\|$, i.e., **minimize** $\|w\|$ or, the simpler $\frac{1}{2}\|w\|^2$
- We arrive to the **hard margin** SVC primal problem is

$$\min_{w,b} \frac{1}{2}\|w\|^2 \text{ s.t. } y^p(w \cdot x^p + b) \geq 1$$

Cover's Theorem

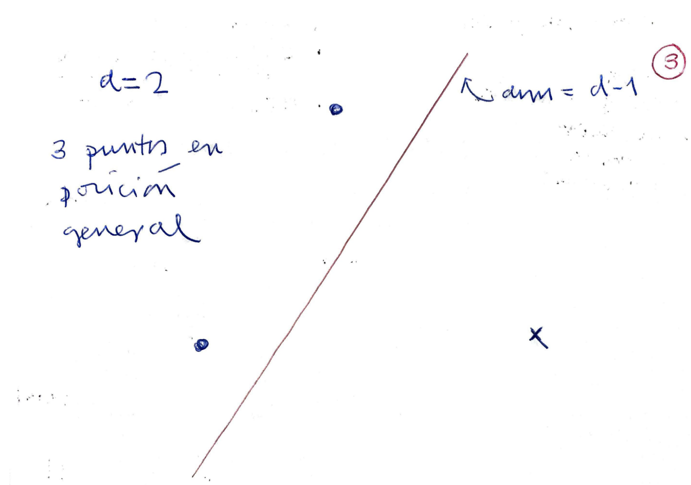
- SVMs are simple and elegant, but also linear
- But, will linear SVM classifiers powerful enough?
- Or, alternatively, **are linearly solvable classification problems frequent enough?**

- Answer: No, because of **Cover's Theorem**
- The patterns in a size N sample S with dimension d are said to be in **general position** if no $d+1$ points are in a $(d-1)$–dimensional hyperplane
- Then, *if* $N \le d+1$, *all 2–class problems on* S *are linearly separable and if* $N > d+1$, *the number of linearly separable samples in general position is*

$$2\sum_{i=0}^{d}\binom{N-1}{i}$$

Points in General Position

- Consider $d=2$, $3=d+1$ points and a $1=d-1$-dimensional hyperplane



Points Not in General Position

- Consider now $d=2$ and $3=d+1$ points **not** on a $1=d-1$-dimensional hyperplane (i.e., a line)

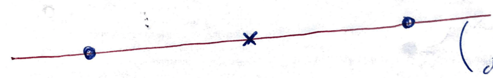
Are Linearly Separable Problems Frequent?

- Our current SVM classifiers will be useful if linearly separable 2–class problems are frequent enough
- It is relatively easy to show that for $N \gg d+1$

$$2\sum_{i=0}^{d}\binom{N-1}{i} \le 2(d+1)\binom{N-1}{d} \le 2\frac{d+1}{d!}N^d \lesssim N^d$$

$d = 2$ 2

$2+1=3$ puntos <u>no</u> en posición general



(dim = 1

- On the other hand, the **total number of two–class problems** over a sample of size N is 2^N
- And $\frac{N^d}{2^N} \to 0$ very fast when $N \to \infty$
- Since in many practical problems we will have $N \gg d$, **essentially all such 2–class problems won't be linearly separable**
- And our current SVMs will be useless on them

Slacks

- What can we do?
- First step: make room for non linearly separable problems

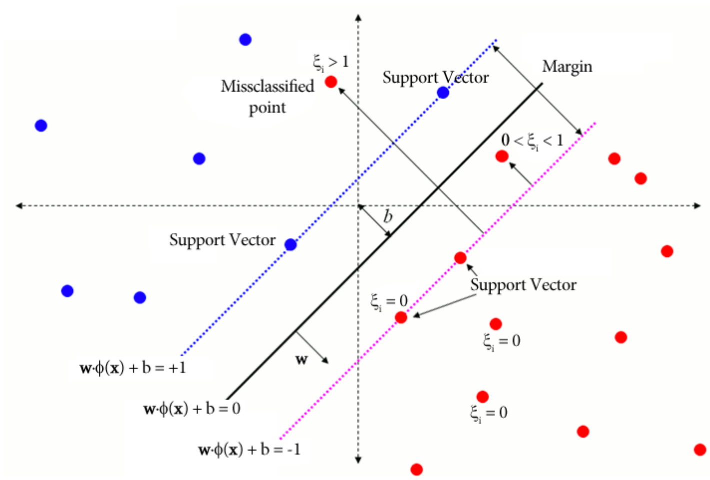
Linear SVMs for Non Linear Probems

- Thus we no longer require perfect classification but **allow for slacks or even errors in some patterns**
- More precisely, we relax the previous requirement $y^p(w \cdot x^p + b) \geq 1$ to

$$y^p(w \cdot x^p + b) \geq 1 - \xi_p$$

  where we impose a new constraint $\xi_p \geq 0$
- Notice that if $\xi_p \geq 1$, x^p will not be correctly classfied
- Thus, we allow for defective clasification but we also **penalize** it

L_k **Penalty SVMs**



- New primal problem: for $K \geq 1$ consider the cost function

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + \frac{C}{K}\sum \xi_p^K$$

now subject to $y^p(w \cdot x^p + b) \geq 1 - \xi_p$, $\xi_p \geq 0$

- Simplest choice $K = 2$: L_2 (i.e., square penalty) SVMs
  - Easy to work out but usually worse models that are not sparse
- Usual (and best) choice $K = 1$
  - We will concentrate on it
- Notice that if $C \to \infty$ we recover the previous slack-free approach

The L_1 Primal Problem

- The soft margin or L_1 SVC primal problem is

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum \xi_p$$

subject to $y^p(w \cdot x^p + b) \geq 1 - \xi_p$, $\xi_p \geq 0$

  - Notice that the loss and the constraints are convex and a Slater point will exists
  - Thus w^*, b^*, ξ^* will be a minimum iff it is a KKT point
  - However, we will pursue duality to solve it
- The L_1 Lagrangian is here

$$L(w, b, \xi, \alpha, \beta) = \frac{1}{2}\|w\|^2 + C\sum \xi_p - \\ \sum \alpha_p \left[y^p(w \cdot x^p + b) - 1 + \xi_p\right] - \sum \beta_p \xi_p$$

with $\alpha_p, \beta_p \geq 0$

Reorganizing the Lagrangian

- We reorganize the L_1 Lagrangian as

$$
\begin{aligned}
L(w, b, \xi, \alpha, \beta) \;&=\; w \cdot \left(\frac{1}{2} w - \sum \alpha_p y^p \; x^p \right) + \\
& \sum \xi_p (C - \alpha_p - \beta_p) - b \sum \alpha_p y^p + \\
& \sum \alpha_p
\end{aligned}
$$

- To get the dual function we solve $\nabla_w L = 0$, $\frac{\partial L}{\partial b} = 0$, $\frac{\partial L}{\partial \xi_p} = 0$
- The w and b partials yield

$$
w = \sum \alpha_p y^p x^p, \;\; \sum \alpha_p y^p = 0
$$

  - The b term drops from the Laplacian and the w term simplifies
  - Moreover, once we get the optimal α^*, we can also get the **optimal** $w^* = \sum_p \alpha_p^* y^p x^p$

The L_1 SVM Dual

- From $\frac{\partial L}{\partial \xi_p} = C - \alpha_p - \beta_p = 0$ we see that

$$
C = \alpha_p + \beta_p,
$$

  - The ξ_p terms also drop from the Laplacian
- Substituting things back into the Lagrangian we arrive at the L_1 dual function

$$
\begin{aligned}
\Theta(\alpha, \beta) \;&=\; \sum_p \alpha_p - \frac{1}{2} w \cdot \sum \alpha_p y^p x^p \\
&=\; \sum_p \alpha_p - \frac{1}{2} \alpha^\tau Q \alpha
\end{aligned}
$$

subject to $\sum_p \alpha_p y^p = 0, \alpha_p + \beta_p = C$, plus $\alpha_p \geq 0, \beta_p \geq 0$ (and both $\leq C$)

Simplifying the L_1 Dual

- In fact, we can drop β
  - Notice that we already have that $\Theta(\alpha, \beta) = \Theta(\alpha)$
  - It is also clear that the constraints on α, β can be reduced to $0 \leq \alpha_p \leq C$
- Thus, we get a much simpler version of the dual problem

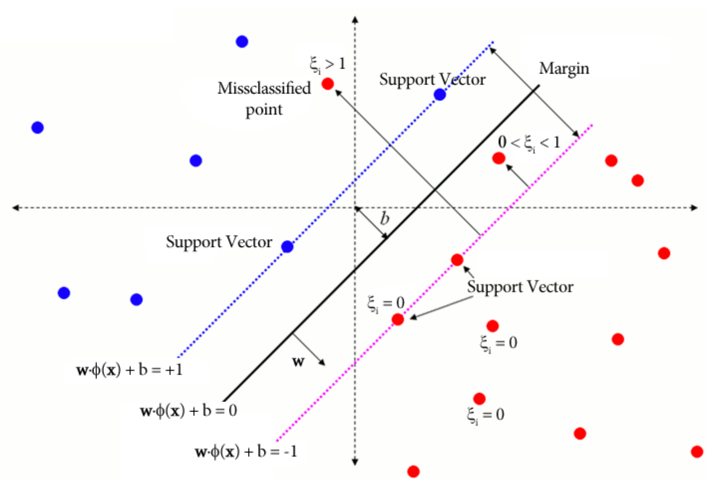
$$
\min_\alpha \;\; \frac{1}{2} \alpha^\tau Q \alpha - \sum_p \alpha_p
$$

subject to $\sum \alpha_p y^p = 0$, $0 \leq \alpha^p \leq C$, $1 \leq p \leq N$

- This is a constrained minimization problem with simple **box** constraints and a harder **linear** one $\sum \alpha_p y^p = 0$

Relevant and Irrelevant Samples

- Recall our previous picture



- We can expect some patterns to influence the final model but others to be irrelevant

KKT Conditions for L_1 SVMs

- The complementary slackness conditions are now

$$\begin{aligned} \alpha_p^* \left[y^p(w^* \cdot x^p + b^*) - 1 + \xi_p^* \right] &= 0 \\ \beta_p^* \xi_p^* &= 0 \end{aligned}$$

  - And also recall that $\alpha_p^* + \beta_p^* = C$

- Now, if $\xi_p^* > 0$, then $\beta_p^* = 0$ and, therefore, $\alpha_p^* = C$
  - We say that such an x^p is **at bound**

- Also, if $0 < \alpha_p^* < C$, then $\beta_p^* > 0$ and $\xi_p^* = 0$
  - Thus, if $0 < \alpha_p^* < C$, $y^p(w^* \cdot x^p + b^*) = 1$ and x^p lies in one of the **support hyperplanes** $w^* \cdot x + b^* = \pm 1$
  - We can obtain $b^* = y^p - w^* \cdot x^p$ from any supporting x^p
  - If needed, we can then derive $\xi_p^* > 0$, since then $\alpha_p^* = C$ and

$$\xi_p^* = 1 - y^p(w^* \cdot x^p + b^*)$$

Projected Gradient Descent

- For homogeneous SVMs without the b term, the linear constraint disappears
- We can then solve the homogeneous dual by **projected gradient descent**
- The gradient of Θ is just
$$\nabla\Theta = Q\alpha - \mathbf{1}$$
with $\mathbf{1}$ the all ones vector and we can solve it by **projected gradient descent**
- Projected (i.e., clipped) descent:
  - At step t update first α^t to α' as $\alpha'_p = \alpha^t_p - \rho\left((Q\alpha^t)_p - 1\right)$ for an appropriate step ρ
  - And then clip α' as $\alpha^{t+1}_p = \min\{\max\{\alpha'_p, 0\}, C\}$
- But usually homogeneous SVMs give poorer results
  - And if sample size N is large, Q will be huge and each step very costly

The SMO Algorithm

- The simplest way to handle the equality constraint is
  - Start with an α^0 that verifies it
  - Update α^t to $\alpha^{t+1} = \alpha^t + \rho_t d^t$ with a direction d^t that also verifies it
  - Then $\sum_p \alpha^{t+1}_p y^p = \sum_p \alpha^t_p y^p + \rho_t \sum_p d^t_p y^p = 0$
- Simplest choice: select L_t, U_t so that $d^t = y^{L_t} e_{L_t} - y^{U_t} e_{U_t}$ is a maximal **descent direction**
- Since $\nabla\Theta(\alpha^t)\cdot d^t = y^{L_t}\nabla\Theta(\alpha^t)_{L_t} - y^{U_t}\nabla\Theta(\alpha^t)_{U_t}$, the straightforward choice is
$$L_t = \arg\min_p y^p \nabla\Theta(\alpha^t)_p, \quad U_t = \arg\min_q y^q \nabla\Theta(\alpha^t)_q$$
- This is the basis of the **Sequential Minimal Optimization** (SMO), the standard algorithm for the general case
  - Effective but also quite costly

Good Option, But ...

- L_1 SVMs are (relatively) **sparse**, i.e., the number of non–zero multipliers should be $\ll N$
- The bound $\alpha^*_p = C$ for $\xi^*_p > 0$ limits the effect of not correctly classified patterns
- And usually L_1 SVMs are much better than, say, L_2 SVMs
- But still **they are linear ...**
- We must thus somehow **introduce some kind of non–linear processing for SVMs to be truly effective**
  - To do so, one observes that SVCs and SMO only require to compute dot products
  - This and the **Kernel Trick** leads to the very powerful kernel SVMs
  - Although probably not for big data problems as their training cost is $\Omega(N^2)$