CS480/680: Introduction to Machine Learning
Lecture 03: Optimization Basics

Yaoliang Yu

University of Waterloo

May 19, 2020

Vector space

- This is our universe for most ML problems.
- Allow us to use linear algebra / calculus tools.

Euclidean space

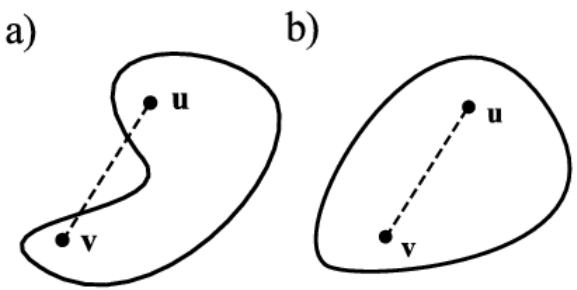
- $\mathbf{v} \in \mathbb{R}^d$ is identified with a d-tuple: $\mathbf{v} = (v_1, v_2, \ldots, v_d)$.
- Addition and multiplication are element-wise
- $\mathbf{u} + \mathbf{v} = (u_1 + v_1, u_2 + v_2, \ldots, u_d + v_d)$;
- $\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$;
- $\mathbf{0} = (0, 0, \ldots, 0)$, $\mathbf{v} + \mathbf{0} = \mathbf{v}$;
- $-\mathbf{v} = (-v_1, -v_2, , \ldots, -v_d)$, $\mathbf{v} - \mathbf{v} = \mathbf{0}$;
- $a(b\mathbf{v}) = (ab)\mathbf{v}$;
- $a\mathbf{v} = (av_1, av_2, \ldots, av_d)$;
- $a(\mathbf{u} + \mathbf{v}) = a\mathbf{u} + a\mathbf{v}$;
- $(a + b)\mathbf{v} = a\mathbf{v} + b\mathbf{v}$.

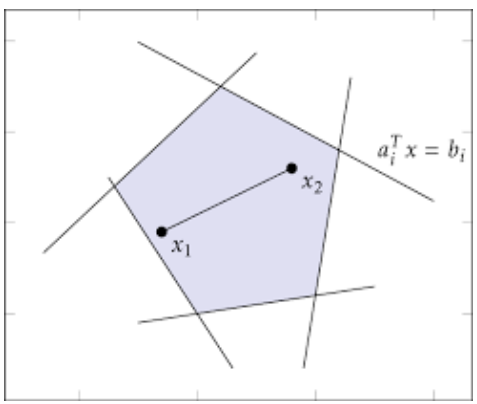
Convex set

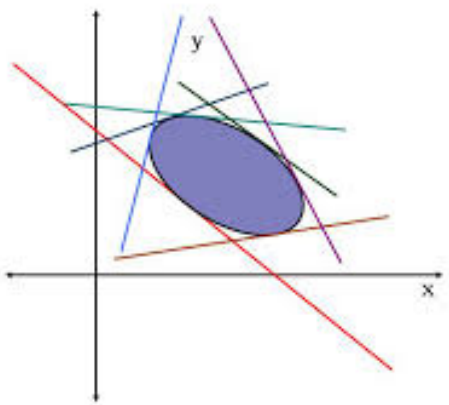
- A point set $C \subseteq \mathbb{R}^d$ is called convex if

$$\forall \mathbf{x}, \mathbf{z} \in C, \ [\mathbf{x}, \mathbf{z}] := \underbrace{\{\lambda \mathbf{x} + (1-\lambda)\mathbf{z} : \lambda \in [0,1]\}}_{\text{convex combination}} \subseteq C.$$

- Intersection of convex sets is convex. Unions are usually not.
- Finite intersection of halfspaces is called polyhedron.
- Any (closed) convex set is an (*infinite*) intersection of halfspaces.





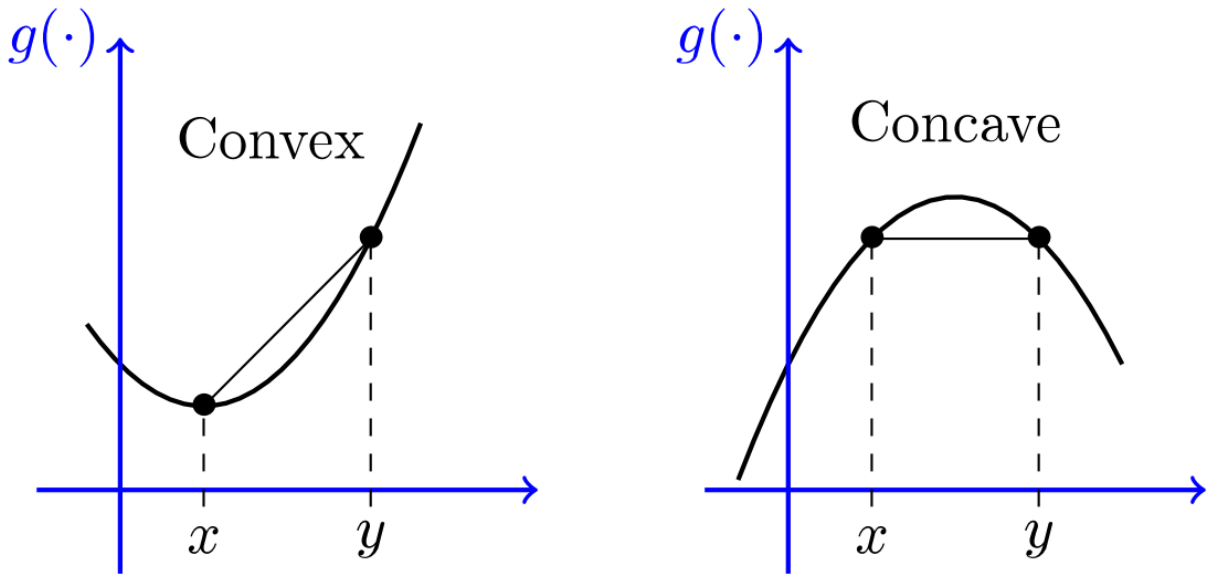


Convex function

- A function $f : C \to \mathbb{R}$ is called convex if
  - domain C is a convex set,
  - for all $\mathbf{x}, \mathbf{z} \in C$, for all $\lambda \in (0, 1)$, Jensen's inequality holds:

$$f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{z}) \leq \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{z}).$$

- We call f *strictly* convex iff equality holds only when $\mathbf{x} = \mathbf{z}$.
- A function f is (strictly) concave iff $-f$ is (strictly) convex.



Verifying convexity

- Let $f : \mathbb{R}^d \to \mathbb{R}$ be twice differentiable. Then f is convex iff its Hessian $\nabla^2 f$ is always positive semidefinite. If the Hessian is always positive definite, then f is strictly convex.

Example

$f(x) = x^4$ is strictly convex but $f''(0) = 0$.

- convex function has increasing derivative along any direction $\mathbf{d}$, starting from any point $\mathbf{x}$.

Calculus of convexity

- Epigraph of any function $\operatorname{epi} f := \{(\mathbf{x}, t) \in C \times \mathbb{R} : f(\mathbf{x}) \leq t\}$.
- f is a convex function iff $\operatorname{epi} f$ is a convex set!
- Any norm is convex;
- If f and g are convex, then for any $\alpha, \beta \geq 0$, $\alpha f + \beta g$ is also convex; (what about $-f$?)
- If $f : \mathbb{R}^d \to \mathbb{R}$ is convex, then so is $\mathbf{w} \mapsto f(A\mathbf{w} + \mathbf{b})$;
- If f_t is convex for all $t \in T$, then $f := \max_{t \in T} f_t$ is convex;
- If $f(\mathbf{x}, t)$ is jointly convex in $\mathbf{x}$ and t, then $\mathbf{x} \mapsto \min_{t \in T} f(\mathbf{x}, t)$ is convex;
- If $f : C \to \mathbb{R}$ is convex, then the perspective function $g(\mathbf{x}, t) := tf(\mathbf{x}/t)$ is convex on $C \times \mathbb{R}_{++}$;

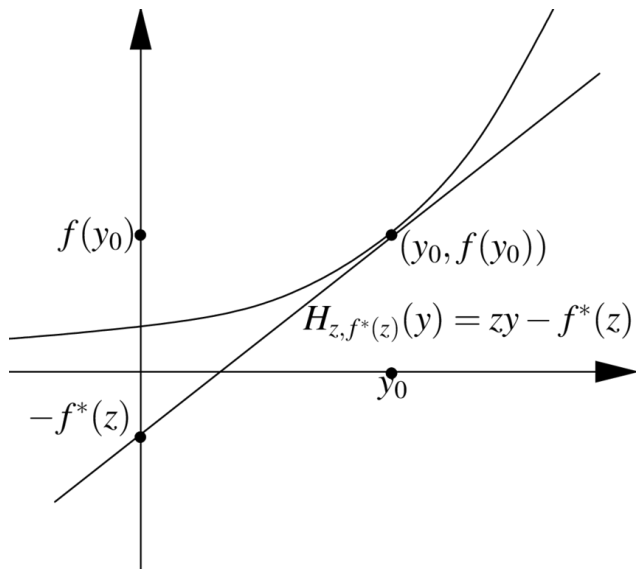
Fenchel conjugate function

The Fenchel conjugate of any function f is:

$$f^*(\mathbf{x}^*) := \max_{\mathbf{x}} \langle \mathbf{x}, \mathbf{x}^* \rangle - f(\mathbf{x}).$$

- According to one of the calculus rules, f^* is always convex.
- If $\operatorname{dom} f$ is closed and f is continuous & convex:

$$f^{**} := (f^*)^* = f.$$



Optimization

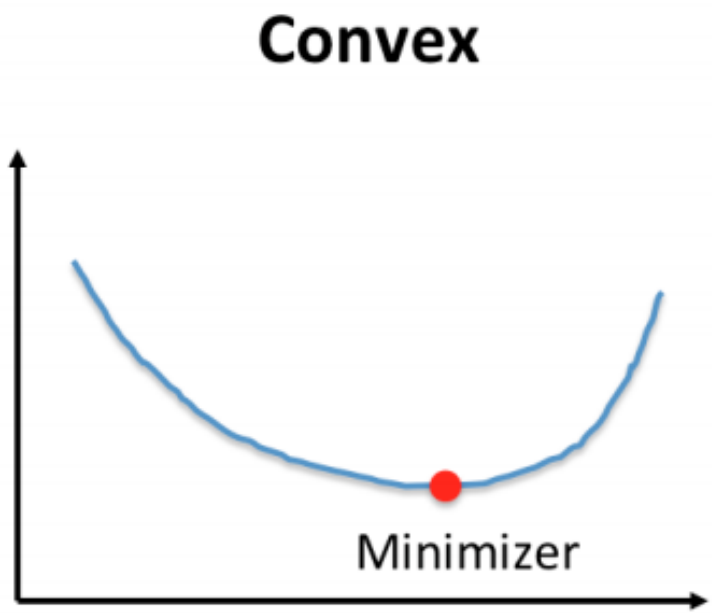
Consider a function $f : \mathbb{R}^d \to \mathbb{R}$, we are interested in the minimization problem:

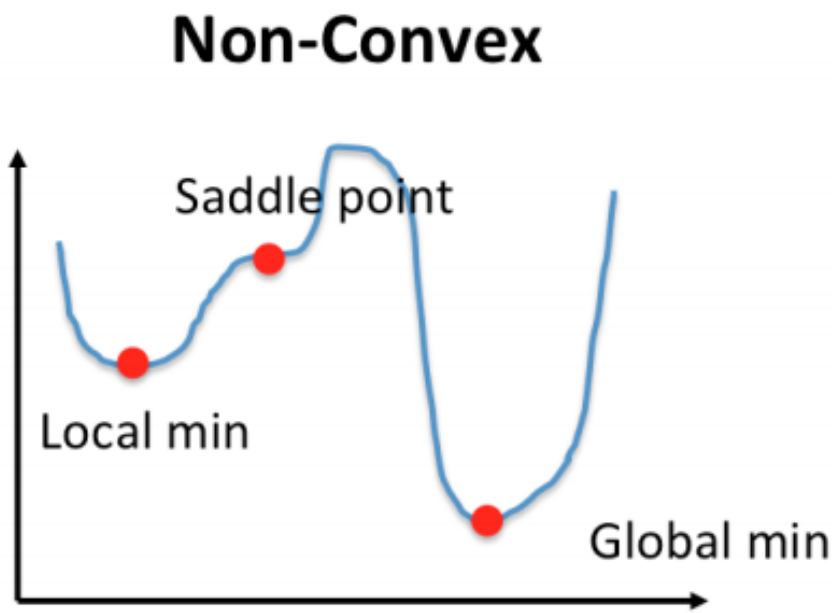
$$\mathfrak{p}_* := \min_{\mathbf{x} \in C} \; f(\mathbf{x}),$$

where $C \subseteq \mathbb{R}^d$ represents the constraints that $\mathbf{x}$ *must* satisfy.

- The minimum value $\mathfrak{p}_*$ is an extended real number in $[-\infty, \infty]$ (where $\mathfrak{p}_* = \infty$ iff $C = \emptyset$).
- When $\mathfrak{p}_*$ is finite, any feasible $\mathbf{x}_* \in C$ such that $f(\mathbf{x}_*) = \mathfrak{p}_*$ is called a (global) minimizer, in notation $\mathbf{x}_* \in \operatorname{argmin}_{\mathbf{x} \in C} \; f(\mathbf{x})$.
- Minimum value always exists while minimizers may not!
- When $\mathbf{x}_*$ minimizes f over a (small) neighborhood in C, we call it local minimizer.
- Global is local and the converse is true under convexity.

The importance of convexity





Properties of minimizing/maximizing

- If $\mathbf{x}$ is a local (global) minimizer (maximizer) of f, then it is a local (global) minimizer (maximizer) of $g(f)$ for any increasing function $g : \mathbb{R} \to \mathbb{R}$. And vice versa if g is *strictly* increasing.

- $\mathbf{x}$ is a local (global) minimizer of f iff $\mathbf{x}$ is a local (global) minimizer of $\lambda f + c$ for any $\lambda > 0$ and $c \in \mathbb{R}$.
- $\mathbf{x}$ is a local (global) minimizer (maximizer) of a positive function f iff it is a local (global) minimizer (maximizer) of $\log f$.

- $\mathbf{x}$ is a local (global) minimizer (maximizer) of a positive function f iff it is a local (global) maximizer (minimizer) of $1/f$.
- $\mathbf{x}$ is a local (global) minimizer of f iff $\mathbf{x}$ is a local (global) maximizer of $-f$.

The epigraph trick

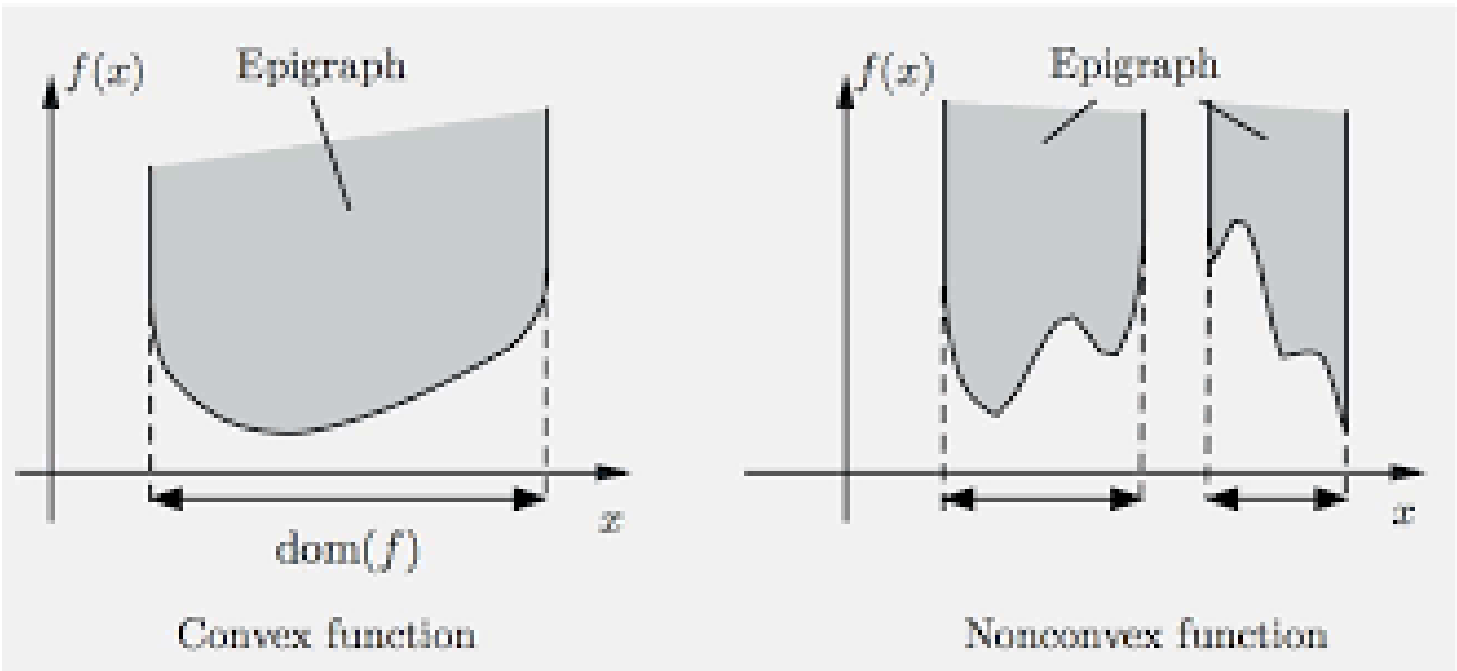
Often, we rewrite the optimization problem

$$\min_{\mathbf{x}\in C} \; f(\mathbf{x})$$

as the equivalent one:

$$\min_{(\mathbf{x},t)\in \operatorname{epi} f \cap C\times\mathbb{R}} \; t,$$

where the newly introduced variable t is jointly optimized with $\mathbf{x}$.



Gradient and Hessian

For a *smooth* function $f : \mathbb{R}^d \to \mathbb{R}$, its

- gradient $\nabla f = (\partial_1 f, \ldots, \partial_d f) \in \mathbb{R}^d$.
- Hessian $\nabla^2 f \in \mathbb{R}^{d \times d}$ with $[\nabla^2 f]_{ij} = \partial_i \partial_j f = \partial_j \partial_i f$.
- match dimensions; gradient always has same size as input $\mathbf{x}$.

Example

Consider the quadratic function $f(\mathbf{x}) = \mathbf{x}^\top Q \mathbf{x} + \mathbf{p}^\top \mathbf{x} + \alpha$:

- input $\mathbf{x} \in \mathbb{R}^d$.
- $Q \in \mathbb{R}^{d \times d}, \mathbf{p} \in \mathbb{R}^d, \alpha \in \mathbb{R}$ are given constants.
- $\nabla f = (Q + Q^\top)\mathbf{x} + \mathbf{p} \in \mathbb{R}^d$.
- $\nabla^2 f = Q + Q^\top \in \mathbb{R}^{d \times d}$.
- memorize them!

Fermat's necessary condition

A necessary condition for $\mathbf{x}$ to be a local minimizer of a smooth function $f : \mathbb{R}^d \to \mathbb{R}$ is

$$\nabla f(\mathbf{x}) = \mathbf{0}.$$

(Such points are called stationary, a.k.a. critical.) If f is convex, then the necessary condition is also sufficient.

Similarly, at a local minimizer we necessarily have $\nabla^2 f(\mathbf{x}) \succeq \mathbf{0}$.

- Fermat's condition gives us a goal in optimization: how do we verify the (sub)optimality of candidate solution $\mathbf{x}$?
- There cannot be any constraints!

The difficulty of satisfying a constraint

Consider the trivial problem:

$$\min_{x \geq 1} \; x^2,$$

which (clearly) admits a unique minimizer $x_\star = 1$.

However, if we ignore the constraint $x \geq 1$ and set the derivative to zero we would obtain $x = 0$, which does not satisfy the constraint!

- We can apply Fermat's condition only when there is no constraint.
- We will introduce the Lagrangian to "remove" constraints.

Algorithm of feasible direction

Input: $\mathbf{x}_0 \in \operatorname{dom} f$
for $t = 0, 1, \ldots$ **do**
  choose direction $\mathbf{d}_t$ // e.g. $\limsup_{t\to\infty} \langle \mathbf{d}_t, \nabla f(\mathbf{x}_t) \rangle > 0$
  choose step size $\eta_t > 0$
  $\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \mathbf{d}_t$ // update
  $\mathbf{x}_{t+1} = \mathsf{P}_C(\mathbf{x}_{t+1})$ // optional projection step
end

- Apply Taylor's expansion:

$$f(\mathbf{x}_{t+1}) = f(\mathbf{x}_t - \eta_t \mathbf{d}_t) = f(\mathbf{x}_t) - \eta_t \langle \mathbf{d}_t, \nabla f(\mathbf{x}_t) \rangle + o(\eta_t),$$

- If $\langle \mathbf{d}_t, \nabla f(\mathbf{x}_t) \rangle > 0$ and η_t is small, then $f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t)$, i.e. the algorithm is descending

Choices

- Gradient Descent (GD): $\mathbf{d}_t = \nabla f(\mathbf{x}_t)$;
- Newton: $\mathbf{d}_t = [\nabla^2 f(\mathbf{x}_t)]^{-1} \nabla f(\mathbf{x}_t)$;
- Stochastic Gradient Descent (SGD): $\mathbf{d}_t = \xi_t$, $\mathsf{E}(\xi_t) = \nabla f(\mathbf{x}_t)$.

- Cauchy's rule $\eta_t \in \operatorname{argmin}_{\eta \geq 0} f(\mathbf{x}_t - \eta \mathbf{d}_t)$.
- Curry's rule $\eta_t = \inf\{\eta \geq 0 : f'(\mathbf{x}_t - \eta \mathbf{d}_t) = 0\}$.
- Constant rule: $\eta_t \equiv \eta > 0$, e.g. $\eta \in (0, 1/\|\nabla^2 f\|_{\mathrm{sp}})$.
- Summable rule: $\sum_t \eta_t = \infty, \sum_t \eta_t^2 < \infty$, e.g. $\eta_t = O(1/t)$.
- Diminishing rule: $\sum_t \eta_t = \infty, \lim_t \eta_t = 0$, e.g. $\eta_t = O(1/\sqrt{t})$.

Some subtleties

- $f(\mathbf{x}_t)$ typically converges, but $\mathbf{x}_t$ itself may not.
- $f(\mathbf{x}_t)$ may not converge to a stationary point even when $f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t)$!
- even when $f(\mathbf{x}_t)$ converges to a stationary point, it may not be a local minimizer. Randomness helps!

Lagrangian

Consider the canonical optimization problem:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^d} \; & f(\mathbf{x}) \\ \text{s.t. } & \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \\ & \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned}$$

where $f : \mathbb{R}^d \to \mathbb{R}$, $\mathbf{g} : \mathbb{R}^d \to \mathbb{R}^n$, and $\mathbf{h} : \mathbb{R}^d \to \mathbb{R}^m$.

- Cannot apply Fermat's condition due to the constraints.

Introduce the Lagrangian multipliers (a.k.a. dual variables) $\boldsymbol{\mu} \in \mathbb{R}^n_+$, $\boldsymbol{\nu} \in \mathbb{R}^m$ to move constraints into the Lagrangian:

$$L(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\nu}) := f(\mathbf{x}) + \boldsymbol{\mu}^\top \mathbf{g}(\mathbf{x}) + \boldsymbol{\nu}^\top \mathbf{h}(\mathbf{x}).$$

We can now rewrite the original problem as the fancy min-max problem:

$$\mathfrak{p}_\star := \min_{\mathbf{x}\in\mathbb{R}^d} \max_{\boldsymbol{\mu}\geq\mathbf{0},\boldsymbol{\nu}} \; L(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\nu}).$$

Swapping the min with max

The Lagrangian dual simply swaps the order of min and max:

$$\mathfrak{d}^\star := \max_{\boldsymbol{\mu}\geq \mathbf{0}, \boldsymbol{\nu}} \min_{\mathbf{x}\in\mathbb{R}^d} L(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\nu}).$$

(Here $\mathfrak{d}$ stands for dual, while $\mathfrak{p}$ stands for primal.)

- The dimension of the dual variable $\boldsymbol{\mu}$ is the number of inequality constraints.
- The dimension of $\boldsymbol{\nu}$ is the number of equality constraints.
- Both are different from the dimension of the (primal) variable $\mathbf{x}$.
- There is no constraint on $\mathbf{x}$ in the dual problem, implicit or explicit!

- Weak duality: $\min_{\mathbf{x}} \max_{\mathbf{y}} \; f(\mathbf{x}, \mathbf{y}) \quad \geq \quad \max_{\mathbf{y}} \min_{\mathbf{x}} \; f(\mathbf{x}, \mathbf{y})$
- Strong duality, e.g. equality attained, relies on convexity and some regularity.

The dual problem

The inner minimization in the dual can usually be solved in closed-form:

$$\mathfrak{X}(\boldsymbol{\mu}, \boldsymbol{\nu}) := \operatorname*{argmin}_{\mathbf{x} \in \mathbb{R}^d} \; L(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\nu}).$$

Plugging any minimizer $\mathfrak{X}(\boldsymbol{\mu}, \boldsymbol{\nu})$ back we obtain the dual problem:

$$\max_{\boldsymbol{\mu} \in \mathbb{R}^n_+, \boldsymbol{\nu} \in \mathbb{R}^m} \; L(\mathfrak{X}(\boldsymbol{\mu}, \boldsymbol{\nu}); \boldsymbol{\mu}, \boldsymbol{\nu}).$$

- The original problem had complicated constraints $\mathbf{g}(\mathbf{x}) \leq 0$.
- The dual problem has only 1 simple nonnegative constraint $\boldsymbol{\mu} \geq 0$.
- Why not try to solve the Lagrangian dual instead, then recover a primal solution $\mathfrak{X}(\boldsymbol{\mu}, \boldsymbol{\nu})$ above!

- Good idea but see some caveats in the note.

KKT conditions

- primal feasibility: $\mathbf{g}(\mathbf{x}) \leq \mathbf{0},\ h(\mathbf{x}) = \mathbf{0}$;
- dual feasibility: $\boldsymbol{\mu} \geq \mathbf{0}$;
- stationarity:

$$\nabla f(\mathbf{x}) + \sum_{i=1}^{n} \mu_i \nabla g_i(\mathbf{x}) + \sum_{j=1}^{m} \nu_j \nabla h_j(\mathbf{x}) = \mathbf{0};$$

- complementary slackness: $\langle \boldsymbol{\mu}, \mathbf{g}(\mathbf{x}) \rangle = 0$.
  - combined with primal and dual feasibility, we have in fact

$$\forall i = 1, \ldots, n, \quad \mu_i g_i(\mathbf{x}) = 0.$$

- KKT conditions are necessary; they give us a goal in optimization.
- They are also sufficient if g_i are convex, h_j are affine and some regularity condition holds.

(Kernel) ridge regression

Recall ridge regression:

$$\min_{\mathbf{w},\mathbf{z}} \frac{1}{2}\|\mathbf{z}\|_2^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2^2$$

$$\text{s.t. } X\mathbf{w} - \mathbf{y} = \mathbf{z},$$

where we *introduced* an "artificial" constraint (and variable $\mathbf{z}$).

- Derive the Lagrangian dual:

$$\max_{\boldsymbol{\alpha}} \min_{\mathbf{w},\mathbf{z}} \frac{1}{2}\|\mathbf{z}\|_2^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \boldsymbol{\alpha}^\top(X\mathbf{w} - \mathbf{y} - \mathbf{z}),$$

- Applying Fermat's condition to the inner minimization problem:

$$\mathbf{w}_\star = -X^\top\boldsymbol{\alpha}/\lambda, \quad \mathbf{z}_\star = \boldsymbol{\alpha}$$

- Plugging it back in (and simplify) we obtain the dual:

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2\lambda}\|X^\top\boldsymbol{\alpha}\|_2^2 - \boldsymbol{\alpha}^\top\mathbf{y} - \frac{1}{2}\|\boldsymbol{\alpha}\|_2^2$$

- Applying Fermat's condition again we obtain:

$$\boldsymbol{\alpha}^\star = -(XX^\top/\lambda + I)^{-1}\mathbf{y}.$$

Gradient descent ascent (GDA)

$$\min_{\mathbf{x}\in\mathsf{X}} \max_{\mathbf{y}\in\mathsf{Y}} f(\mathbf{x},\mathbf{y})$$

Input: $(\mathbf{x}_0,\mathbf{y}_0)\in \operatorname{dom} f\cap \mathsf{X}\times\mathsf{Y}$

$s_{-1}=0, (\bar{\mathbf{x}}_{-1},\bar{\mathbf{y}}_{-1})=(\mathbf{0},\mathbf{0})$ `// optional`

for $t=0,1,\ldots$ **do**

choose step size $\eta_t>0$

$\mathbf{x}_{t+1}=\mathsf{P}_{\mathsf{X}}[\mathbf{x}_t-\eta_t\nabla_{\mathbf{x}} f(\mathbf{x}_t,\mathbf{y}_t)]$ `// GD on minimization`

$\mathbf{y}_{t+1}=\mathsf{P}_{\mathsf{Y}}[\mathbf{y}_t+\eta_t\nabla_{\mathbf{y}} f(\mathbf{x}_t,\mathbf{y}_t)]$ `// GA on maximization`

$s_t=s_{t-1}+\eta_t$

$(\bar{\mathbf{x}}_t,\bar{\mathbf{y}}_t)=\frac{s_{t-1}(\bar{\mathbf{x}}_{t-1},\bar{\mathbf{y}}_{t-1})+\eta_t(\mathbf{x}_t,\mathbf{y}_t)}{s_t}$ `// averaging`

end

The last step incrementally performs averaging:

$$(\bar{\mathbf{x}}_t,\bar{\mathbf{y}}_t)=\sum_{k=1}^{t}\eta_k(\mathbf{x}_k,\mathbf{y}_k)/\sum_{k}\eta_k$$

Variations of GDA

- use different step sizes on $\mathbf{x}$ and $\mathbf{y}$;
- use $\mathbf{x}_{t+1}$ in the update on $\mathbf{y}$ (or vice versa);
- use stochastic gradients in both steps;
- after every update in $\mathbf{x}$, perform k updates in $\mathbf{y}$ (or vice versa);