Convex Optimization in Machine Learning and Computational Finance Lecture 10: Black-box Methods 2

Dr. Michel Baes, Pr. Patrick Cheridito

RiskLab / ETH Zürich

Quick recall of last week's lecture

- General optimization problems cannot be solved.
- Lipschitz continuity is a measure of well-posedness.
- Lower complexity bounds are obtained
  using the black-box model.
- For univariate bounded convex problems,
  nothing beats the bisection method in theory.
- For convex problems with many variables,
  using subgradients (first-order oracle),
  subgradient method is theoretically the best.
- Useful variant: stochastic gradient method.
  - ▷ for problems with large datasets (minibatch strategies)
  - ▷ for stochastic optimization problems

I. Better complexity results for smooth convex problems

General convex problems

$\min_{x \in Q} f(x)$

- $Q \subseteq \mathbb{R}^n$ is a convex set;
- $f : Q \to \mathbb{R}$ is convex;
- desired accuracy on objective's value is ϵ.

Recall: if we have an oracle $x \mapsto (f(x), g(x))$, where $g(x) \in \partial f(x)$ and $x \mapsto \pi_Q(x)$, if

$$|f(x) - f(y)| \leq L||x - y||_2 \quad \forall x, y \in \text{dom} f,$$

we need $\mathcal{O}((||x^* - x_0||_2 L/\epsilon)^2)$ oracle calls.
And there is no hope of being faster on this class.
Note: Other versions exist for other norms
(mirror-descent methods)

Smooth convex problems

$\min_{x \in Q} f(x)$

- $Q \subseteq \mathbb{R}^n$ is a convex set;
- $f : Q \to \mathbb{R}$ is convex and differentiable;
- desired accuracy on objective's value is ϵ.

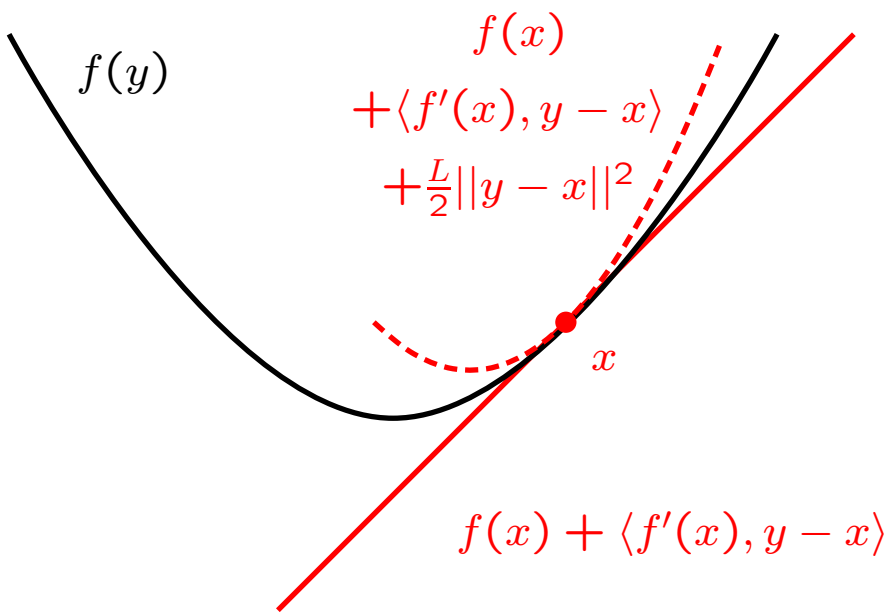
New assumption: f has a *Lipschitz continuous gradient*, or is *smooth*, i.e. $\exists L > 0$ for which:

$$||f'(y) - f'(x)||_* \leq L||y - x|| \quad \forall x, y \in \text{dom} f.$$

Note: We do not restrict ourselves to Euclidean norm here.

- We call such functions *L-smooth w.r.t. $||\cdot||$* or *L-smooth* (if the norm is clear from the context).

An intuitive illustration of gradient Lipschitz continuity



And the proof

Suppose f is strictly convex. Then:

$$||f'(y) - f'(x)||_* \leq L||y - x|| \quad \forall x, y \in \mathrm{dom} f.$$

$$\Leftrightarrow \quad f(y) - f(x) - \langle f'(x), y - x\rangle \leq \frac{L}{2}||y - x||^2 \quad \forall x, y \in \mathrm{dom} f.$$

Proof:

$$\begin{aligned} \text{" } \Rightarrow \text{ " :} \quad & |f(y) - f(x) - \langle f'(x), y - x\rangle| \\ = \; & \left| \int_0^1 \langle f'(x + t(y - x)) - f'(x), y - x\rangle dt \right| \\ \leq \; & \int_0^1 ||f'(x + t(y - x)) - f'(x)||_* dt ||y - x|| \leq \frac{L}{2}||y - x||^2. \end{aligned}$$

And the proof

Suppose f is strictly convex. Then:

$$||f'(y) - f'(x)||_* \leq L||y - x|| \quad \forall x, y \in \mathrm{dom} f.$$

$$\Leftrightarrow \quad f(y) - f(x) - \langle f'(x), y - x\rangle \leq \frac{L}{2}||y - x||^2 \quad \forall x, y \in \mathrm{dom} f.$$

Proof: "$\Leftarrow$": Let us fix $x, y \in \mathrm{dom} f$ and $s \in \mathbb{R}^n$. First:

$$f(y) + f_*(s) = f(y) + \sup_{z \in \mathrm{dom} f} \langle s, z\rangle - f(z) = f(y) + \sup_{z \in \mathbb{R}^n} \langle s, z\rangle - f(z)$$

$$\geq \sup_{z \in \mathbb{R}^n} \langle s, z\rangle - \langle f'(y), z - y\rangle - \frac{L}{2}||y - z||^2 = \langle s, y\rangle + \frac{||s - f'(y)||_*^2}{2L}.$$

Let now $s := f'(x)$. Since f is strictly convex, the only maximizer of $\langle f'(x), z\rangle - f(z)$ is $z := x$. Thus $f_*(f'(x)) = \langle f'(x), x\rangle - f(x)$, and:

$$f(y) \geq f(x) + \langle f'(x), y - x\rangle + ||f'(x) - f'(y)||_*^2/(2L).$$

As $f(y) \leq f(x) + \langle f'(x), y - x\rangle + \frac{L}{2}||y - x||^2$, we get the desired inequality.

A new step-size rule for the gradient method

- We are using the Euclidean norm.
- Assume f is L-smooth w.r.t. $||\cdot||_2$ and strictly convex.

The method says: $x_{k+1} = x_k - h_k f'(x_k)$; thus:

$$f(x_{k+1}) = f(x_k - h_k f'(x_k)) \leq f(x_k) - h_k ||f'(x_k)||_2^2 + \frac{L h_k^2}{2} ||f'(x_k)||_2^2.$$

Minimizing the right-hand side in h_k, we get $h_k^* = 1/L$.

- With this choice, $f(x_{k+1}) \leq f(x_k) - ||f'(x_k)||_2^2/(2L)$.
- A useful inequality:

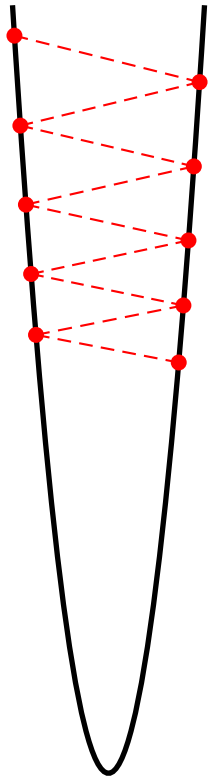
$$f(y) \geq f(x) + \langle f'(x), y - x \rangle + ||f'(x) - f'(y)||_2^2/(2L)$$

implies that $\langle f'(z), z - x^* \rangle \geq ||f'(z)||_2^2/L$
(**Why?** Hint: take $(x, y) = (z, x^*)$ then $(x, y) = (x^*, z)$).

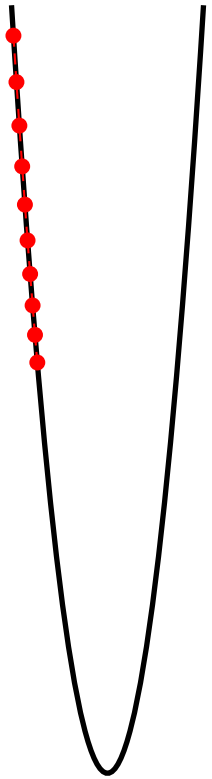
The intuitive meaning of L

What happens if the step-size is not $1/L$

Step-size: $1/L + \delta$



Step-size: $1/L - \delta$



difference between step-sizes: $2\delta = 0.06$

The convergence is better

$$x_{k+1} = x_k - f'(x_k)/L.$$

We use $x \mapsto ||x - x^*||_2^2$ as potential. First, the potential decreases:

$$\begin{aligned} ||x_{k+1} - x^*||_2^2 &= ||x_k - x^*||_2^2 - 2\langle f'(x_k), x_k - x^*\rangle/L + ||f'(x_k)||_2^2/L^2 \\ &\leq ||x_k - x^*||_2^2 - 2||f'(x_k)||_2^2/L^2 + ||f'(x_k)||_2^2/L^2 \\ &= ||x_k - x^*||_2^2 - ||f'(x_k)||_2^2/L^2. \end{aligned}$$

Second, $f(x_k) - f^* \leq \langle f'(x_k), x_k - x^*\rangle \leq ||f'(x_k)||_2 ||x_k - x^*||_2$
$\leq ||f'(x_k)||_2 ||x_0 - x^*||_2 = D||f'(x_k)||_2$, with $D := ||x_0 - x^*||_2$.

Third, we know that $f(x_{k+1}) \leq f(x_k) - ||f'(x_k)||_2^2/(2L)$.

Let $\Delta_k := f(x_k) - f^*$. We have $\Delta_{k+1} \leq \Delta_k - \Delta_k^2/(2LD^2)$, or:

$$\frac{1}{\Delta_{k+1}} \geq \frac{1}{\Delta_k} + \frac{\Delta_k}{\Delta_{k+1}} \frac{1}{2LD^2} \geq \frac{1}{\Delta_k} + \frac{1}{2LD^2} \geq \frac{1}{\Delta_0} + \frac{k+1}{2LD^2} \geq \frac{k+1}{2LD^2}.$$

Thus $f(x_N) - f^* \leq 2LD^2/N$, and the complexity is $\mathcal{O}(LD^2/\epsilon)$
(instead of $\mathcal{O}(L_0^2 D^2/\epsilon^2)$ for non-smooth problems).

Note: For constrained problems, we get the same result:
do the usual $||\pi_Q(x - x^*)||_2 \leq ||x - x^*||_2$

What about lower bounds?

The best lower bound Yudin and Nemirovski could find is:

$$N = \min\left\{(n-1)/2, D\sqrt{3L/2\epsilon}/4\right\}$$

iterations for a first-order oracle method.

Note: Surprisingly enough, the "hard" function is quadratic.

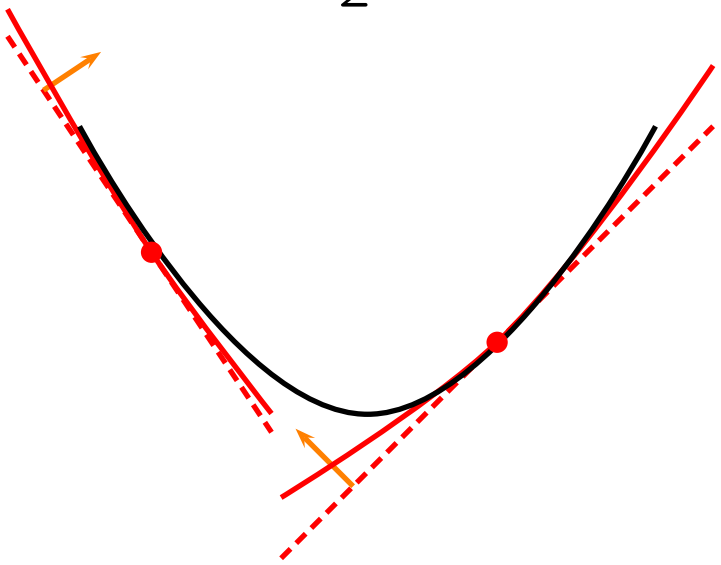
(For a proof, see Nesterov's book, Section 2.1.2.)

Are there better methods
for L-smooth convex functions?

Yes, an optimal method exists for strongly convex problems

Extra assumption: f is *μ-strongly convex*:

$$f(y) - f(x) - \langle f'(x), y - x \rangle \geq \frac{\mu}{2}||x - y||^2 \quad \forall x, y \in \mathrm{dom} f.$$



Note: If f is L-smooth w.r.t. $||\cdot||$,
then f_* is $1/L$-strongly convex w.r.t. $||\cdot||_*$.

Spectacular effect of a few extra lines of code: the optimal method

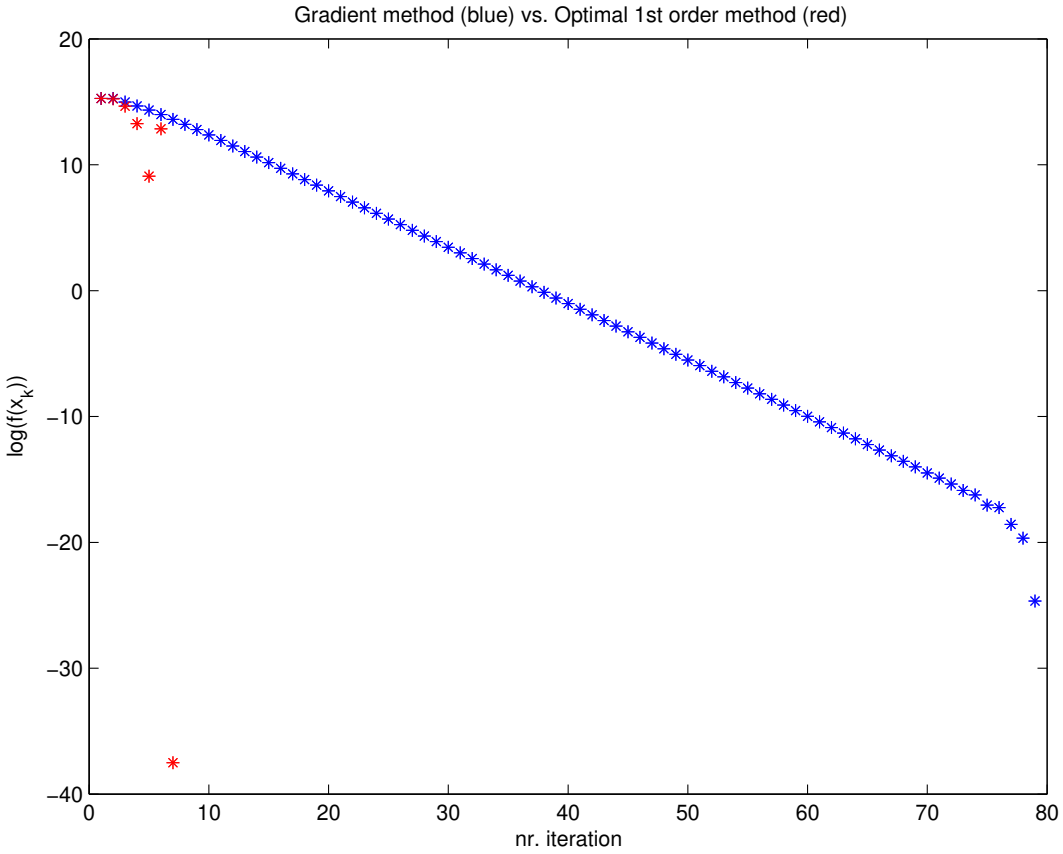
- $\min_x f(x)$, f **μ-strongly convex**, $\mu > 0$.
- f differentiable, and $||f'(y) - f'(x)||_2 \leq L||y - x||_2$.
- Oracle: $x \mapsto (f(x), f'(x))$.

$y_0 = x_0, \beta := (\sqrt{L} - \sqrt{\mu})/(\sqrt{L} + \sqrt{\mu})$
for all $k \geq 0$
$x_{k+1} = y_k - f'(y_k)/L$
$y_{k+1} = x_{k+1} + \beta(x_{k+1} - x_k)$

Theorem 1 (Nesterov, 1983) $f(x_k) - f(x^*) \leq 4LD^2/(k+2)^2$
Justification + Proof: 12 (hard) pages.
(See Section 2.2 in Nesterov's book).

- A complicated variant exists for $\mu = 0$ with the **same** convergence speed. (3 sequences must be constructed).

Numerically, it often converges much faster



First order methods

Problem assumptions	**Resisting oracle**	**Method**
Convex, non-smooth Lip. continuous (L_0)	partial max + norm, $\Omega\left(\left(\frac{L_0 D}{\epsilon}\right)^2\right)$	Subgradient $\mathcal{O}\left(\left(\frac{L_0 D}{\epsilon}\right)^2\right)$
Convex, constrained, L-smooth Strongly convex (μ)	quadratic fct, $\Omega\left(D\sqrt{\frac{L}{\epsilon}}\right)$	Gradient $\mathcal{O}\left(\frac{LD^2}{\epsilon}\right)$ Optimal $\mathcal{O}\left(D\sqrt{\frac{L}{\epsilon}}\right)$

II. Gradient methods for non-convex problems

We can only hope to find a stationary point

$$\boxed{\min_{x\in\mathbb{R}^n} f(x)} \quad (\mathcal{P})$$

$\star$ $f:\mathbb{R}^n\to\mathbb{R}$ is differentiable,
L-smooth w.r.t. $||\cdot||_2$,
and $f^* := \min f(x) > -\infty$.

We know that there is no hope to find the global optimum without incredible luck in a reasonable time.

So we must content ourselves to find a *stationary point* x_s (i.e. $f'(x_s)=0$) approximately: $||f'(\widehat{x})||_2 < \epsilon$.

This goal is reachable by the gradient method for an appropriate choice of the step-size.

With "optimal" step-sizes, the method converges to a stationary point

The best step-size gradient method

Start from x_1. For $k \geq 1$ compute $x_{k+1} := x_k - h_k f'(x_k)$, where $h_k := \arg\min_{h \geq 0} f(x_k - h f'(x_k))$.

$$\begin{aligned} f(x_{k+1}) &= \min\{f(x_k - h f'(x_k)) : h \geq 0\} \\ &\leq \min\{f(x_k) - h||f'(x_k)||_2^2 + Lh^2||f'(x_k)||_2^2/2 : h \geq 0\} \\ &= f(x_k) - ||f'(x_k)||_2^2/(2L). \end{aligned}$$

Summing up from $k := 1$ to $k := N$, and rearranging,

$$f(x_1) - f(x_{N+1}) \geq \sum_{k=1}^{N} \frac{||f'(x_k)||_2^2}{2L} \geq \min_{1 \leq k \leq N} \frac{N||f'(x_k)||_2^2}{2L}.$$

If $N \geq 2L(f(x_1) - f^*)/\epsilon^2$, then $\min_{k \leq N} ||f'(x_k)||_2 \leq \epsilon$.

The method also converges for easier step-size strategies*

Solving $\arg\min_{h\in\mathbb{R}+} f(x_k - hf'(x_k))$ can be very hard.
A vast literature is devoted to step-size strategies.

- **Armijo-Goldstein step-size conditions**

Let $\phi(h) := f(x_k - hf'(x_k))$ and $1 < \gamma$. We require:

$\phi(h_k) \leq f(x_k) - h_k||f'(x_k)||_2^2/5$ [Sufficient decrease]

$\phi(\gamma h_k) \geq f(x_k) - \gamma h_k||f'(x_k)||_2^2/5$ [Forbid too small h_k]

Then $\min_{k\leq N} ||f'(x_k)||_2 < \epsilon$ for $N = \mathcal{O}(\gamma L(f(x_i) - f^*)/\epsilon^2)$.

- **Methods with multiple step-sizes**

  The step-size is different for every component:
  $x_{k+1,i} = x_{k,i} - h_{k,i}f_i'(x_k)$ for all $1 \leq i \leq n$.
  Example: Adam's Method (Convergence speed: ???)
  $1/h_{k,i}$ estimate the Lipschitz constant of $t \mapsto f'(x_k + te_i)$.
  (Also, instead of $f'(x_k)$, Adam's method uses
  a moving average of previous gradients).

III. Methods with a second-order oracle

Using the best norm: Newton method

Recall the reformulation of gradient method:
$x_{k+1} = x_k - hf'(x_k)$ iff

$$x_{k+1} = \arg\min_y f(x_k) + \langle f'(x_k), y - x_k\rangle + \frac{1}{2h}||y - x_k||^2.$$

Consider:

$$\widehat{x}_{k+1} = \arg\min_y f(x_k) + \langle f'(x_k), y - x_k\rangle + \frac{1}{2}\langle f''(x_k)(y - x_k), y - x_k\rangle$$

(We have replaced the matrix I/h in the norm by $f''(x)$)

$$\widehat{x}_{k+1} = x_k - f''(x_k)^{-1} f'(x_k)$$

Newton's method has drawbacks

- Oracle: $x \mapsto (f(x), f'(x), f''(x))$.
- The Hessian $f''(x)$ should be invertible.
- Cannot be applied as such for constrained problems.
- Expensive if $f''(x)$ is dense.

What about convergence?

Convergence is much faster

Suppose that $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ satisfies:

- f is twice differentiable.
- $f''(x^*) \succeq lI \succ 0$.
- The Hessian f'' is Lipschitz continuous w.r.t. $||\cdot||_2$:
  $\forall\, x, y \; |||f''(x) - f''(y)|||_2 \leq M||x - y||_2$,
  where $|||A|||_2 := \sup\{||Ax||_2 : ||x||_2 \leq 1\}$.

Exercise: $|||A|||_2 = \max_i\{|\lambda_i(A)|\}$.

Theorem 2 (Kantorovich)
If $||x_0 - x^*||_2 \leq 2l/3M$, the iterates x_k of Newton's method are well-defined and:

$$||x_{k+1} - x^*||_2 \leq \frac{M||x_k - x^*||_2^2}{2(l - M||x_k - x^*||_2)} \leq \frac{3M}{2l}||x_k - x^*||_2^2.$$

Kantorovich's proof

Note: A critical analysis of this result will lead Nesterov and Nemirovski to revolutionize convex optimization.

Proof: Let $r_k := ||x_k - x^*||_2$ for every k for which $f''(x_k)$ is invertible.

First, iterates are well-defined when $r_k < l/M$ (OK for $k := 0$).
Indeed, $f''(x_k) - f''(x^*) \succeq -M||x_k - x^*||_2 I$, and $f''(x_k) \succeq (l - Mr_k)I_n$.

Second, we have when $r_k < l/M$:

$$\begin{aligned} x_{k+1} - x^* &= x_k - x^* - f''(x_k)^{-1} f'(x_k) \\ &= f''(x_k)^{-1} \left(\int_0^1 [f''(x_k) - f''(x^* + t(x_k - x^*))](x_k - x^*) dt \right) \\ \Rightarrow \quad r_{k+1} &\leq |||f''(x_k)^{-1}|||_2 \left(\int_0^1 |||f''(x_k) - f''(x^* + t(x_k - x^*))|||_2 dt \right) r_k \\ &\leq |||f''(x_k)^{-1}|||_2 \frac{M}{2} r_k^2 \leq \frac{M r_k^2}{2(l - Mr_k)}. \end{aligned}$$

Note that $r_{k+1} < r_k$ when $r_k < 2l/(3M)$, because then $\frac{Mr_k^2}{2(l-Mr_k)} < r_k$.
Therefore $r_k < l/M$ for every k, and the iterates are well-defined.

Newton's method has drawbacks

- Oracle: $x \mapsto (f(x), f'(x), f''(x))$.
- The Hessian $f''(x)$ should be invertible.
- Cannot be applied as such for constrained problems.
- Expensive if $f''(x)$ is dense.
- Convergence is faster, but only local.

Is Newton's method practical at all? Of course!

The theory of Self-Concordant functions [Nesterov and Nemirovski] shows quite exactly how Newton's Method can be used at best.

An improvement on Newton's Method [Nesterov, Polyak]

For functions with a Lipschitz continuous Hessian, a *cubic regularization* allows to get rid of $f''(x^*) \succeq lI$ and $||x_0 - x^*||_2 \leq 2l/3M$ to guarantee convergence.

IV. Back to Convex Optimization

The internal contradiction of subgradient methods: Towards mirror descent methods

What is a gradient?

$f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ continuous, $d \in \mathbb{R}^n$.

- The function f is *differentiable in* $x \in \mathrm{dom} f$
  if the following limit exists:

$$\nabla f(x)[d] = \lim_{t \downarrow 0} \frac{f(x+td) - f(x)}{t},$$

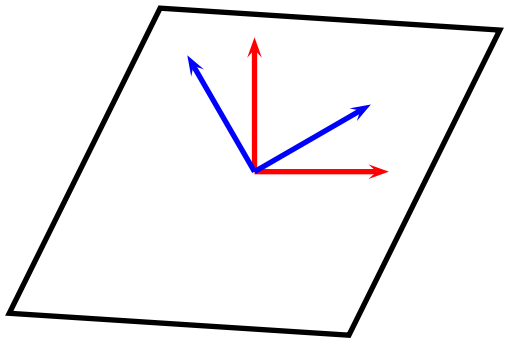
  and if $d \mapsto \nabla f(x)[d]$ is linear; (aka *Gâteaux differentiablility*)
- The *gradient* of f at x is the vector $f'(x)$
  satisfying $\langle f'(x), d \rangle = \nabla f(x)[d]$
  for the **dot scalar product** (exists by Riesz's Theorem)

A gradient depends on the particular scalar product used.
Nothing justifies the a priori use of the dot product.

An analogy:

is the canonical basis always the best?

Suppose your feasible set is a diamond.

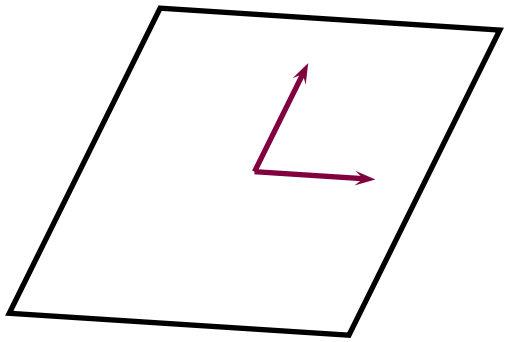


Obviously, the blue basis
is better than the red one.

An analogy:

is the canonical basis always the best?

Suppose your feasible set is a diamond.



Obviously, the blue basis
is better than the red one.
But the best is the purple one,
which is orthogonal w.r.t.
a **different** scalar product.

So where is the contradiction, and how do we resolve it?

$$x_{k+1} = x_k - h_k f'(x_k).$$

x_k is a point in $\mathbb{R}^n$, while $f'(x_k)$ represents a linear function through a scalar product. It makes no sense to add them.

- The *dual space* E^* of $E := \mathbb{R}^n$ is the set of linear functions $f : E \to \mathbb{R}$.

Rule: combine only elements of E^* with each other, and map the result in E.

Observation: Let V be a smooth convex function $E \to \mathbb{R}$. Then $V_*(s) := \sup_x s(x) - V(x)$ maps $E^* \to \mathbb{R}$, $\nabla V : E \to E^*$, and $\nabla V_* : E^* \to E$.

∇V_* will be our "dual to primal" transformation.

The mirror descent method

$\min_{x \in E} f(x)$

- $f : E \to \mathbb{R}$ is diff. convex;
- oracle: $x \mapsto (f(x), \nabla f(x))$;
- absolute accuracy ϵ.

Here $s_k \in E^*$ and $x_k \in E$.

We have chosen $V : E \to \mathbb{R}$ such that V_* is L-smooth.

$x_0 \in \mathbb{R}^n, s_0 := 0.$
for all $k \geq 0$
$s_{k+1} := s_k - h_k \nabla f(x_k)$
$x_{k+1} := \nabla V_*(s_{k+1})$

As Nemirovski says: *the iterations are actually made "through the mirror E^*" of E.*

- If $V(x) := ||x||_2^2/2$, we recover the gradient method.
- $x_{k+1} = \nabla V_*(s_{k+1})$ iff $s_{k+1} = \nabla V(x_{k+1})$ (see Lecture 5).

The mirror descent method works for every smooth norm

We require that V_* is $\mathcal{L}$-smooth w.r.t. $||\cdot||_*$,
e.g. $V(x) = ||x||^2/2$ and $V_*(s) = ||s||_*^2/2$.

Convergence (Nemirovski): We use as potential $\phi(s) := V_*(s) - s(x^*)$. (What is it for $V(x) = ||x||_2^2/2$?) We have:

$$\begin{aligned}
\phi(s_{k+1}) &= \phi(s_k - h_k \nabla f(x_k)) = V_*(s_k - h_k \nabla f(x_k)) - s_k(x^*) + h_k \nabla f(x_k)[x^*] \\
&\leq V_*(s_k) - h_k \nabla V_*(s_k)[\nabla f(x_k)] + \frac{\mathcal{L} h_k^2}{2} ||\nabla f(x_k)||_*^2 - s_k(x^*) + h_k \nabla f(x_k)[x^*] \\
&= V_*(s_k) - h_k \nabla f(x_k)[x_k] + \frac{\mathcal{L} h_k^2}{2} ||\nabla f(x_k)||_*^2 - s_k(x^*) + h_k \nabla f(x_k)[x^*] \\
&= \phi(s_k) - h_k \nabla f(x_k)[x_k - x^*] + \frac{\mathcal{L} h_k^2}{2} ||\nabla f(x_k)||_*^2.
\end{aligned}$$

Summing up from $k := 0$ to $k := N$, and assuming $||\nabla f(x)||_* \leq L$:

$$\frac{\sum_{k=0}^N h_k f(x_k)}{\sum_{k=0}^N h_k} - f(x^*) \leq \frac{\phi(s_0) - \phi(s_{N+1}) + \sum_{k=0}^N h_k^2 \mathcal{L} L^2/2}{\sum_{k=0}^N h_k}.$$

The mirror descent method works for every smooth norm

Essentially, the proof is the same
as for the subgradient method, and the result as well:

$$\frac{\sum_{k=0}^{N} h_k f(x_k)}{\sum_{k=0}^{N} h_k} - f(x^*) \leq \frac{\phi(s_0) - \phi(s_{N+1}) + \sum_{k=0}^{N} h_k^2 \mathcal{L} L^2/2}{\sum_{k=0}^{N} h_k}$$

Suppose $V_*(s_0) = 0$. As $V_*(s_{N+1}) \geq s_{N+1}(x^*) - V(x^*)$,
we have $\phi(s_0) - \phi(s_{N+1}) \leq V(x^*)$.

Convergence is ensured if $\sum_{k=0}^{N} h_k \to \infty$
and $\sum_{k=0}^{N} h_k^2 < c$ as $N \to \infty$ (cf. subgradient method)

Note 1: Nemirovski proved that if $V(x) = ||x||_p^2/2$,
the best p is $p^* := 1 + \kappa/\log(n)$, with $\kappa \approx 1$.

Note 2: Mirror-descent methods work also
with stochastic gradients (see last lecture)

Betting on horses with a mirror descent method in disguise

Note: What I am about to show is frequently used
to accelerate large-scale algorithms
(e.g. Fast MinCut method [Madry],
Fast Graph Laplacian solver [see Vishnoi], ...),
but also in Finance, even though it is much riskier.

Trying to bet as well as they can

People are gambling on horses (or on CDOs,. . .)

1. Every gambler except you
   chooses a strategy once and forever.
2. You are allowed to change your strategy at every round.

> Eventually, can you hope
> to be as good as the best gambler?

Yes! [Freund and Schapire, ADABoost Algorithm, 1995]

Average loss w.r.t. the best gambler:

$\mathcal{O}(\sqrt{\ln(N)/T})$,

where N is the number of gamblers

T the number of rounds.

The trick: give a score to everyone and bet accordingly

The setting

Let $(\Omega, P, \mathcal{B})$ be a probability space

The loss of gambler j is $l_j : \Omega \to [-\mu, \rho]$

Multiplicative Weight Algorithm

Let w_j be the score of gambler j (initially, $w_j := 1/N$), and $\omega_k \in \Omega$ the outcome on day k.

- If j has a win ($l_j(\omega_k) < 0$), increase w_j

$$w_j := w_j(1+\epsilon)^{-l_j(\omega_k)/\rho}$$

- Otherwise, penalize her:

$$w_j := w_j(1-\epsilon)^{l_j(\omega_k)/\rho}.$$

- At day $k+1$, follow gambler j with probability $w_j / \sum_i w_i$.

Variants and specializations

$$w_j := w_j(1+\epsilon)^{-l_j(\omega_k)/\rho} \text{ if } l_j(\omega_k) < 0$$
$$w_j := w_j(1-\epsilon)^{l_j(\omega_k)/\rho} \text{ if } l_j(\omega_k) \geq 0$$

- **ADABoost** [176.000 Google hits]
  specifies a particular loss l_j and changes ϵ at every k.
- **Hedge Algorithm**
  Select a score function $U_\gamma : [-\mu, \rho] \to \mathbb{R}$ such that:

$$\gamma^{\frac{\mu+t}{\mu+\rho}} \leq U_\gamma(t) \leq 1 - (1-\gamma)\frac{\mu+t}{\mu+\rho}.$$

$$w_j := w_j U_\gamma(l_j(\omega_k))$$

The Hedge Algorithm converges

Algorithm 1

Set $w_{0,i} := 1/n$ *for all* $1 \leq i \leq N$, $p_0 := w_0$.

Choose a function U_γ.

for $k = 0 : T$

Generate or observe a realization $\omega_k \in \Omega$.

$w_{k+1,j} := w_{k,j} U_\gamma(l_j(\omega_k))$. [Reevaluate scores]

$p_{k+1,j} := w_{k+1,j} / \sum_{i=1}^{n} w_{k+1,i}$. [Normalize]

Pick the decision of gambler j *with probability* $p_{k+1,j}$.

Variant if X **is convex:**

Pick the decision $x_k := \sum_{j=1}^{N} p_{k+1,j} x_{k,j}$.

end

The Hedge Algorithm converges

Theorem 3 *[Freund, Shapire, ...] Choose $T \geq 0$ such that*

$$\epsilon := \sqrt{\frac{\rho \ln(N)}{T(2\mu + \rho)}} \leq \frac{1}{2}.$$

Then

$$\underbrace{\sum_{k=0}^{T-1} \sum_{j=1}^{N} \frac{p_{k,j} l_j(\omega_k)}{T}}_{\text{What you get on average}} \leq \underbrace{\min_{i=1,\ldots,N} \left\{ \sum_{k=0}^{T-1} \frac{l_j(\omega_k)}{T} \right\}}_{\text{What the best gets on average}} + \sqrt{\frac{2 \ln(N)}{T}} + \frac{\ln(N)}{T}.$$

But it can converge faster

The Hedge algorithm is a mirror descent method with (very) approximate gradients

$$\phi(x,\omega) := \sum_{j=1}^{n} x_j \ln(U_\gamma(l_j(\omega))), \quad f(x) := E_P[\phi(x,\omega)] \quad \forall \omega \in \Omega, x \in \mathbb{R}^n.$$

$$V(x) = \sum_{j=1}^{n} x_j \log(x_j), \quad \text{use } \frac{\partial \phi(x_k,\omega_k)}{\partial x} \text{ instead of } f'(x_k)$$

...but with a very poor choice of step-sizes.

Taking the right ones, we can improve the complexity and its practical behavior. [Buergisser]

For next week(s)

- **Making Newton's method work**:

Self-concordant functions.

- **Interior-point methods**:

What makes them so efficient? What are their limits?