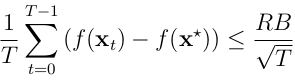
EPFL CS-439 (optim. for ML) spring 2022

template from <https://github.com/dcetin/eth-cs-notes/> (pai-cheatsheet)

## GD basics



#### Lipschitz convex: 1/eps^2

**** 

#### Smooth convex: 1/eps

** **

suff. descent: ****

#### Smooth convex accelerated: 1/sqrt(eps)

Nesterov’s accelerated gradient descent (‘83)

#### Smooth strongly-convex: (L/mu) log(1/eps)

** **

****

## PGD



(second term can be seen as noise, often cancels out)

#### **Lipschitz convex: idem, 1/eps^2**

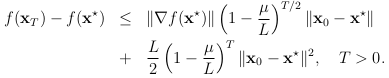
Technically, only need bounded gradient ⊋ lipschitz (X closed)

#### **Smooth convex: 1/eps**

suff. “descent” 

#### **Smooth strongly-convex: (L/mu) log(1/eps)**

square distance to OPT still geom. decreasing, but



### Proximal grad: f=g+h,

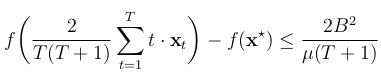
 non-expansive

g smooth, g,h convex, **: **

## Subgradient descent

Lipschitz convex: idem, 1/eps^2

#### **Tame strongly-convex : 1/**eps

Reason for step-size choice: must multiply by t before telescoping

#### **Lower bound (Nesterov): ∃ f B-lipschitz s.t**

for any subgradient method, 

## SGD

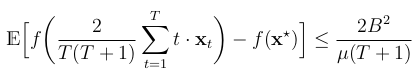
Smoothness never helps! Only bounded gradients (“tame”)

(Smoothness may help when doing variance-reduced SGD)

#### Lipschitz convex: idem, 1/eps^2

Technically, only need bounded stoch. gradients 

#### Tame strongly-convex : 1/eps

(same proof as in subgradient descent with expectations)

#### Mini-batch **reduces variance:**

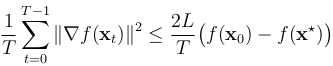
where batch size=m and 

## Nonconvex functions

Def. of smooth is still only an upper bound!

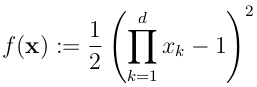
TL;DR: →0 at same rate as for convex

#### Smooth: 1/eps ON AVERAGE

** **

No-overshoot ppty: ∄ critical pt on segment [xt,xt+1]****

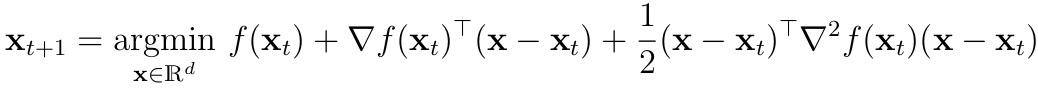
#### The example



Not globally smooth, yet



## Newton’s method

(No step-size) 

**Rk.** Newton’s method is **affine-invariant**.

**Thm.** Suppose ∃ ball around x\* where (for spectral norm)

 and 

Then if x0∊ball, 

**Corr.** If x0∊ball and , 

i.e 

Local quadratic convergence (“double the number of correct digits in each iteration”)

* affine invariant
* converge in 1 step for quadratics

## Quasi-Newton

with H symmetric s.t secant condition:



In 1D, only one secant method

Greenstadt family, of which (L-)BFGS

Newton ∊ Quasi-Newton ⇔ f nondegen. quadratic

## Coordinate descent

PL inequality: 

μ-strongly-convex ⇒ μ-PL

E.g f(x)=x1^2 is 1-PL but not SC

E.g 

#### GD on smooth + PL: (L/mu) log(1/eps)

(Exact same proof as for smooth+SC)

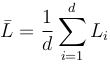
#### Randomized CD

If f is (L, …, L)-coord-wise-smooth and μ-PL, ,



#### Importance sampling CD

If f is (L1, …, Ld)-coord-wise-smooth and μ-PL, stepsize ,

 ; 

E.g f(x)=x1^2 is (2, 0, …, 0)-smooth so L=2 and ˉL=2/d

#### Steepest CD aka Gauss-Southwell

If f is (L, …, L)-coord-wise-smooth and μ-PL, same bound as for Randomized CD

→ strictly worse bound, as per-iteration cost is ~d

If f is (L, …, L)-coord-wise-smooth and μ1-PL **w.r.t l1 norm**,

, 

Rk: μ1-SC w.r.t l1 norm ⇒ μ1-PL w.r.t l1 norm



#### Greedy CD: line-search

May fail: 

**Thm.** If f = g+h, g convex diffble and

 with hi convex

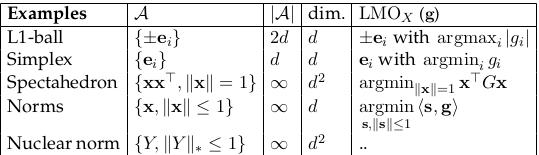
then xt+1=xt ⇒ xt global min of f

## Frank-Wolfe aka conditional gradient



#### Lin. Min. Oracle

If X = conv(A), then LMO\_X(g) ∊ A (“atoms”)

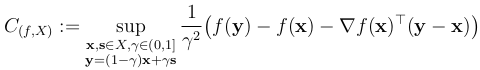


(Spectrahedron: PSD matrices with trace=1;  via eigenvector)

#### Duality gap

g(x) ≥ f(x)-f(x\*) and g(x\*) = 0

#### Curvature constant



If f is L-smooth, then .

Allows to capture that Frank-Wolfe algo is **affine-invariant**.

#### Smooth convex: 1/eps

Analysis is quite different from (S/P/prox)GD, CD, Newton.

**Thm.** If X convex compact, f convex, C\_(f,X) < ∞,

step-size  (indep of params!)

or  or ,

then 

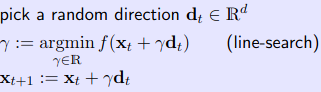
and so 

and so .

**Thm (“cv” of duality gap).** Under the same conditions,

there exists 1≤t≤T s.t 

## Zero-th-order/gradient-free optim

Random search: 

(step-size: no other choice than line-search!)

Convergence rates: same as GD with optimal step-size, with slow-down factor of d

Smooth convex: T < dL/eps

Smooth strongly-convex: T < dL/mu log(1/eps)

## Misc

In finite dim, convex => cont. and difble almost everywhere.

If param e.g **** unknown, use doubling trick.

Using line-searched step-size, can only do better than fixed step-size.

To prove SC => PL: min over y in

f(y) > f(x) + g(x)\*(y-x)+mu/2 |y-x|2

similarly for non-l2 norms

similarly, can prove L-smooth => “lower-PL”

For any convex L-smooth f,



(proof: L-smooth => “lower-PL” on tilted h(x)=f(x)-g(y)\*x)