

# Data Science for Actuaries (ACT6100)

Arthur Charpentier

Rappels # 4.3 (Convex Optimization)

automne 2020

 <https://github.com/freakonometrics/ACT6100/>

# Convex Optimization Problem

$$\min_{\mathbf{x}} \{f(\mathbf{x})\}$$

with  $f$  convex, and differentiable.

---

## Algorithm 1: Gradient Descent

---

- 1 initialization :  $\mathbf{x}^{(0)}$ ;
  - 2 **for**  $t=1,2,\dots$  **do**
  - 3      $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)} - h_t \nabla f(\mathbf{x}^{(t-1)})$
- 

Heuristics: Taylor expansion

$$f(\mathbf{y}) \sim f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2h} \|\mathbf{y} - \mathbf{x}\|^2$$

# Convergence

If  $f$  is convex, differentiable and such that  $\nabla f$  is Lipschitz continuous with some constant  $\gamma > 0$ , i.e.

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq \gamma \|\mathbf{x} - \mathbf{y}\|$$

then if  $h < 1/\gamma$ ,

$$f(\mathbf{x}^t) - f^* \leq \frac{\|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2}{2ht}$$

i.e. gradient descent converges at rate  $1/t$ , or we can find some  $\epsilon$ -suboptimal point in  $1/\epsilon$  iterations.

If  $f$  is non-convex, differentiable and such that  $\nabla f$  is Lipschitz continuous with some constant  $\gamma > 0$ , gradient descent converges at rate  $1/\sqrt{t}$

# Non Differentiable Case

If  $f$  is convex and differentiable

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x})$$

for all  $\mathbf{x}, \mathbf{y} \in \text{dom}(f)$ .

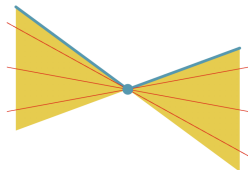
If  $f$  is convex and non-differentiable, for all  $\mathbf{x}$ , there is  $\mathbf{g}$  such that

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top (\mathbf{y} - \mathbf{x})$$

for all  $\mathbf{y} \in \text{dom}(f)$ .

$\mathbf{g}$  is called subgradient at point  $\mathbf{x}$ .

If  $f$  differentiable at  $\mathbf{x}$ ,  $\mathbf{g}$  is unique and  $\mathbf{g} = \nabla f(\mathbf{x})$



# Non Differentiable Case

The set of subgradients of a convex function  $f$  is the subdifferential,

$$\partial f(\mathbf{x}) = \{\mathbf{g} \in \mathbb{R}^n : \mathbf{g} \text{ is a subgradient at } \mathbf{x}\}$$

Note that  $\partial f(\mathbf{x})$  is a convex set, and if  $f$  is differentiable at point  $\mathbf{x}$ ,  $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$

Proposition: for any  $f$ ,  $f(\mathbf{x}^*) = f^*$  if and only if  $\mathbf{0} \in \partial f(\mathbf{x})$ .

# Convex Optimization Problem

$$\min_{\mathbf{x}} \{f(\mathbf{x})\}$$

with  $f$  convex, but nondifferentiable.

---

## Algorithm 2: Subgradient 'Descent'

---

```
1 initialization :  $\mathbf{x}^{(0)}$ ;  
2 for  $t=1,2,\dots$  do  
3    $\mathbf{g}^{(t-1)} \in \partial f(\mathbf{x}^{(t-1)})$ ;  
4    $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)} - h_t \mathbf{g}^{(t-1)}$ 
```

---

Note that it is not necessarily a descent, so pick

$$\mathbf{x}^* = \operatorname{argmin}\{f(\mathbf{x}^{(0)}), f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), \dots\}$$

# From Gradient Descent to Newton's Method

$$\min_{\mathbf{x}} \{f(\mathbf{x})\}$$

with  $f$  convex, twice differentiable.

---

## Algorithm 3: Newton's Method

---

```
1 initialization :  $\mathbf{x}^{(0)}$ ;  
2 for  $t=1,2,\dots$  do  
3    $\mathbf{H}_t \leftarrow \nabla^2 f(\mathbf{x}^{(t-1)})$ ;  
4    $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)} - \mathbf{H}_t^{-1} \nabla f(\mathbf{x}^{(t-1)})$ 
```

---

Instead of

$$f(\mathbf{y}) \sim f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2h} \|\mathbf{y} - \mathbf{x}\|^2$$

use a better quadratic approximation  $-\frac{1}{h} \mathbb{I} \rightarrow H$ ,

$$f(\mathbf{y}) \sim f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2} (\mathbf{y} - \mathbf{x})^\top H (\mathbf{y} - \mathbf{x})$$

## Newton (1685) - Raphson (1690)

Let  $g(\mathbf{x}) = \nabla f(\mathbf{x})$ . Assume that  $g(\mathbf{x} + \vec{\mathbf{u}}) = 0$ , then

$$0 = g(\mathbf{x} + \vec{\mathbf{u}}) \sim g(\mathbf{x}) + \nabla g(\mathbf{x}) \vec{\mathbf{u}}$$

i.e.  $\vec{\mathbf{u}} \sim -\nabla g(\mathbf{x})^{-1} g(\mathbf{x})$ , which yields

$$\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)} + \vec{\mathbf{u}}, \text{ with } \vec{\mathbf{u}} = -\mathbf{H}_t^{-1} \nabla f(\mathbf{x}^{(t-1)})$$

If computing the Hessian matrix  $\mathbf{H}_t$  is complicated, one can approximate  $\mathbf{H}_t$  by some (positive definite) matrix: quasi-Newton. Heuristically, use  $\mathbf{H}'$  close to  $\mathbf{H}_t$ , symmetric, e.g.

$\mathbf{H}' = \mathbf{H}_t + \mathbf{a}\mathbf{u}\mathbf{u}^\top$  (symmetric rank one update) or

$\mathbf{H}' = \mathbf{H}_t + \mathbf{a}\mathbf{u}\mathbf{u}^\top + \mathbf{b}\mathbf{v}\mathbf{v}^\top$  (symmetric rank two update), called  
Broyden Fletcher Goldfarb Shanno (BFGS) method



# Coordinate Descent

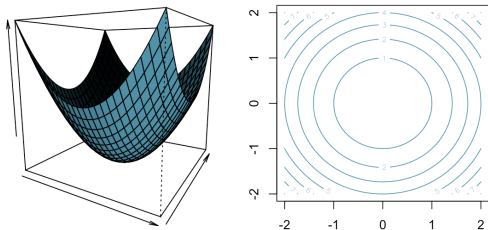
Let  $\{\vec{e}_1, \dots, \vec{e}_n\}$  denote the standard basis in  $\mathbb{R}^n$ ,

$$\vec{e}_i = (0, \dots, 0, 1, 0, \dots, 0) \in \mathbb{R}^n$$

**Proposition** If  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is convex, differentiable,

$$f(\mathbf{x}) \leq f(\mathbf{x} + \delta \vec{e}_i), \forall i \implies f(\mathbf{x}) = \min\{f\}$$

i.e. if we are at a point  $\mathbf{x}$  such that  $f(\mathbf{x})$  is minimized along each coordinate axis, then we have found a global minimizer.

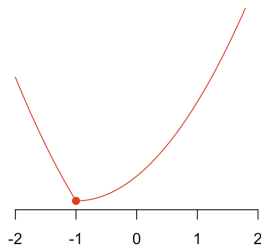
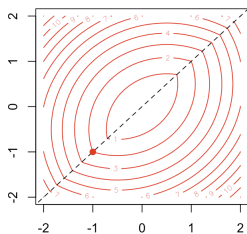
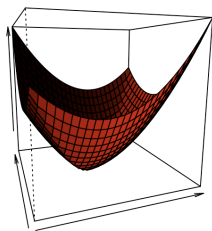


# Coordinate Descent

**Proposition** If  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is convex, but **not differentiable**,

$$f(\mathbf{x}) \leq f(\mathbf{x} + \delta \vec{e}_i), \forall i \not\Rightarrow f(\mathbf{x}) = \min\{f\}$$

i.e. if we are at a point  $\mathbf{x}$  such that  $f(\mathbf{x})$  is minimized along each coordinate axis, then we have **not** found a global minimizer.



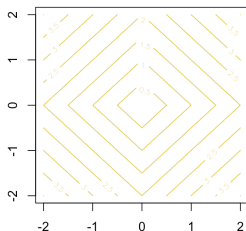
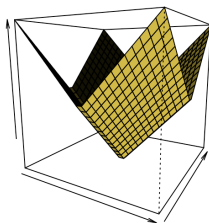
# Coordinate Descent

**Proposition** If  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  can be written

$$f(\mathbf{x}) = g(\mathbf{x}) + \underbrace{\sum_{i=1}^n h_i(\mathbf{x}_i)}_{\text{separable}}, \quad \text{where } \begin{cases} g \text{ convex and differentiable} \\ h_i \text{ convex and non-differentiable} \end{cases}$$

$$f(\mathbf{x}) \leq f(\mathbf{x} + \delta \vec{e}_i), \quad \forall i \implies f(\mathbf{x}) = \min\{f\}$$

i.e. if we are at a point  $\mathbf{x}$  such that  $f(\mathbf{x})$  is minimized along each coordinate axis, then we have found a global minimizer.



# Coordinate Descent

If we want to solve  $\min\{f(\mathbf{x})\}$  for some  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  such that

$$f(\mathbf{x}) = g(\mathbf{x}) + \underbrace{\sum_{i=1}^n h_i(\mathbf{x}_i)}_{\text{separable}}, \quad \text{where } \begin{cases} g \text{ convex and differentiable} \\ h_i \text{ convex and non-differentiable} \end{cases}$$

we can use a **coordinate descent algorithm**

---

## Algorithm 4: Coordinate Dscent

---

```
1 initialization :  $\mathbf{x}^{(0)}$ ;  
2 for  $t=1,2,\dots$  do  
3   for  $j=1,2,\dots,n$  do  
4      $\mathbf{x}_j^{(t)} \leftarrow \operatorname{argmin}\{f(\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_{j-1}^{(t)}, \mathbf{x}_j, \mathbf{x}_{j+1}^{(t-1)}, \dots, \mathbf{x}_n^{(t-1)})\}$ 
```

---

# Gradient vs. Coordinate Descent

Consider the problem  $\min\{f(\beta)\}$  where  $f(\beta) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|^2$

- ▶ Gradient descent,  $\beta \leftarrow \beta + h\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\beta)$
- ▶ Coordinate descent,  $\beta_j \leftarrow \beta_j + \frac{1}{\mathbf{X}_j^\top \mathbf{X}_j} \mathbf{X}_j^\top(\mathbf{y} - \mathbf{X}\beta)$

to go further...

- ▶ Noisy descent

$$\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t-1)} - h_t \nabla f(\mathbf{x}^{(t-1)}) + \varepsilon^{(t-1)}$$

where  $\varepsilon^{(t-1)}$  is some zero-mean Gaussian noise, with decreasing variance.

- ▶ Simulated annealing, genetic algorithms, etc.