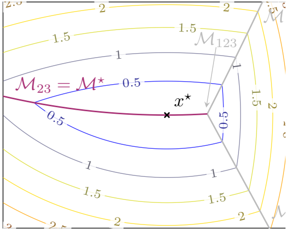
Nonsmooth Optimization at Work

models, geometry, and applications in energy and learning

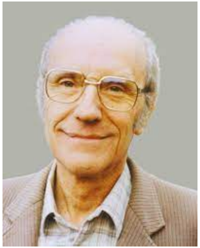
Jérôme MALICK

Journées SMAI-MODE – Lyon – March 2024

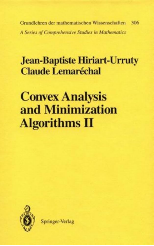
Teasing...



Optimal manifold


J.-J. Moreau

histogram reshaping

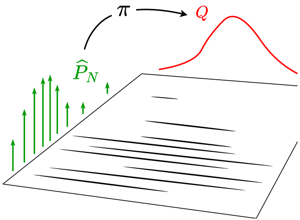

Grundlehren der mathematischen Wissenschaften 306
A Series of Comprehensive Studies in Mathematics

Jean-Baptiste Hiriart-Urruty
Claude Lemaréchal

Convex Analysis
and Minimization
Algorithms II

Springer-Verlag

Hiriart-Urruty LemarechaL



Wasserstein ambiguity


flying pigs

March 27th 2004

12ème journées du groupe MODE

$\mathcal{U}$-Lagrangien et géométrie

Jérôme MALICK[1], **Scott MILLER**[2]

[1] INRIA (Rhône-Alpes)
Montbonnot, 38334 St Ismier
`jerome.malick@inria.fr`

[2] University of California, San Diego
9500 Gilman Dr, m/c 0411, La Jolla, CA 92093-0411
`scott@turbulence.ucsd.edu`

RESUME

La méthode de Newton peut être considérée comme le prototype des algorithmes rapides d'optimisation. Dans cet exposé, nous comparons différentes manières de l'étendre à des problèmes d'optimisation non lisse. Les précisions sur le contenu de l'exposé se trouvent dans [3].

Le cadre de travail est le suivant. On s'intéresse à la minimisation sur $\mathbb{R}^n$ d'une fonction convexe f, et on suppose que le minimum est atteint sur une sous-variété $\mathcal{M}$ par apport à laquelle f est partly-smooth. Introduite dans [2], la partial smoothness exprime essentiellement que la régularité de f est confinée à $\mathcal{M}$. Le problème se reformule comme un problème de minimisation sous contraintes

$$\begin{cases} \min f(x) \\ x \in \mathcal{M}\,. \end{cases}$$

L'objectif est de préciser les liens entre différentes manières adapter la méthode de Newton à ce problème:

– les algorithmes provenant de la théorie du $\mathcal{U}$-Lagrangian de [1],
– les méthodes SQP,
– les méthodes de Newton locales sur $\mathcal{M}$.

- 20 years ago !
- first conf'
- SMAI-MODE 2004
- Le Havre
- nonsmoothness & geometry
- towards Newton methods for minimizing nonsmooth functions

Nonsmooth objective functions are everywhere...

Max functions

$$F(x) = \sup_{u \in U} h(u, x)$$

- robust optimization, stochastic optimization, Benders decomposition
- Lagrangian relaxations of combinatorial problems

Nonsmooth regularization

$$F(x) = f(x) + g(x)$$

- image/signal processing, inverse problems
- sparsity-inducing regularizers in machine learning

Nonsmooth composition

$$F(x) = g \circ c(x)$$

- risk-averse optimization, eigenvalue optimization
- deep learning: nonsmooth activation, implicit layers

Probability functions

$$F(x) = \mathbb{P}\big(h(x, \xi) \leqslant 0\big)$$

- optimization under uncertainty, energy optimization

So what ?...

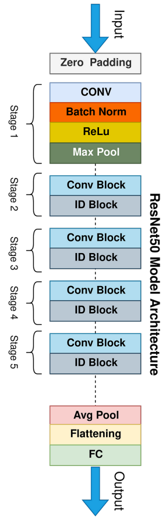
Is nonsmoothness really important ? useful ?

Why not just ignoring it ?

- **Ex:** nonsmooth deep learning
  with RELU, max-pooling or implicit layers
- Just apply SGD with back-prog
- Or just apply quasi-Newton with (sub)gradients

Why not smoothing it ?

- Smoothing by (inf-)convolution (e.g. Moreau regularization)
- Smoothings by overparameterization, ad hoc, or...



So what ?...

Is nonsmoothness really important ? useful ?

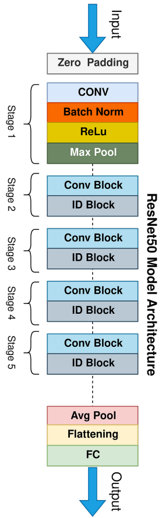
Why not just ignoring it ?

- **Ex:** nonsmooth deep learning
  with RELU, max-pooling or implicit layers
- Just apply SGD with back-prog
- Or just apply quasi-Newton with (sub)gradients

Why not smoothing it ?

- Smoothing by (inf-)convolution (e.g. Moreau regularization)
- Smoothings by overparameterization, ad hoc, or...

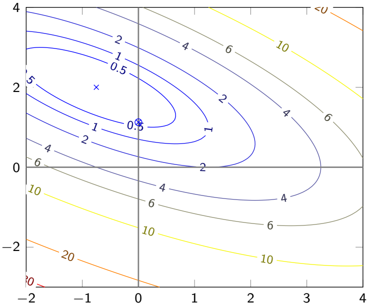
My point: nonsmoothness is relevant !



Example: ℓ_1-regularized least-squares (1/2)

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{2}\|Ax-y\|^2 + \lambda\|x\|_1 \qquad \text{(LASSO)}$$

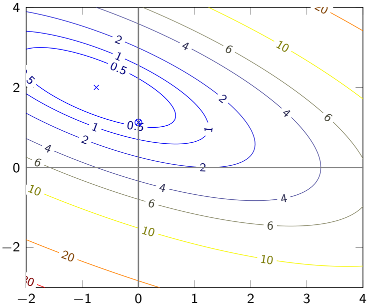
Illustration (on an instance with $d=2$)

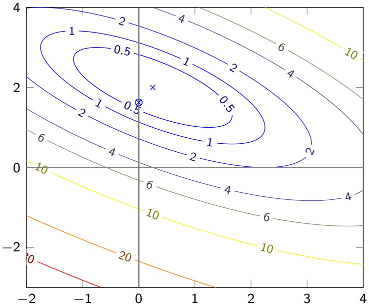


Example: ℓ_1-regularized least-squares (1/2)

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{2}\|Ax - y\|^2 \ + \ \lambda\|x\|_1 \qquad \text{(LASSO)}$$

Illustration (on an instance with $d = 2$)





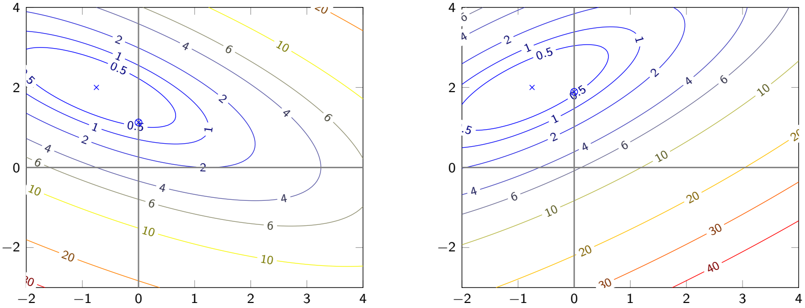
the support of optimal solutions is stable under small perturbations

Nonsmoothness traps solutions in low-dimensional manifolds

Example: ℓ_1-regularized least-squares (1/2)

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{2}\|Ax - y\|^2 + \lambda\|x\|_1 \qquad \text{(LASSO)}$$

Illustration (on an instance with $d = 2$)

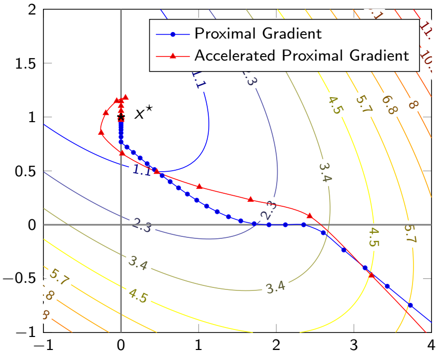


the support of optimal solutions is stable under small perturbations

Nonsmoothness traps solutions in low-dimensional manifolds

Example: ℓ_1-regularized least-squares (2/2)

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{2}\|Ax-y\|^2 + \lambda\|x\|_1 \qquad \text{(LASSO)}$$

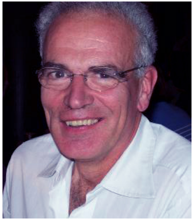


(proximal-gradient) algorithms produce iterates...

...that eventually have the same support as the optimal solution

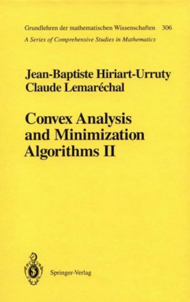
$\boxed{\text{Nonsmoothness attracts (proximal) algorithms}}$

Remark: smooth but stiff problems

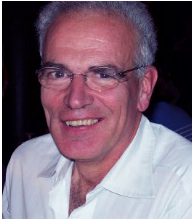


J.-B. Hiriart-Urruty   C. Lemaréchal

"There is no clear cut between functions that are smooth and functions that are not.
In-between there is a rather fuzzy boundary of stiff functions"

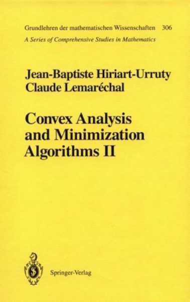


Remark: smooth but stiff problems



J.-B. Hiriart-Urruty C. Lemaréchal

"There is no clear cut between functions that are smooth and functions that are not.
In-between there is a rather fuzzy boundary of stiff functions"



In sharp contrast with smoothing-like approaches:

- Toy example from the book (Section VIII.3.3): for a smooth problem, run usual algorithms

  bundle (nonsmooth) $\gg$ (smooth) gradient, conj. grad., quasi-Newton

- Real-life example in energy optimization :
  - problem of managment of reservoirs : smooth
  - state-of-the-art algos to solve it : nonsmooth

$\boxed{\text{Nonsmoothness can help, even for (difficult) smooth problems}}$

This talk: advocacy for nonsmooth optimization

Nonsmoothness is sometimes useful, sometimes unavoidable – and always nice-looking

Goals of this talk:

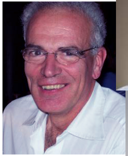
- Illustrations of its role, its geometry...
- One math spotlight on the proximal operator
- 2 spotlights on applications:
  - in industry : electricity generation
  - in learning : towards robustness and fairness
- High level: underline ideas, duality, models...

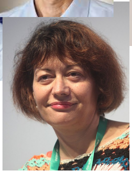
  No theorems ! No algorithms ! No references !
- modest goals + a personal view

Nonsmooth optimization at work: Outline

1. Spotlight 1: Do you know all about prox ?
2. Spotlight 2: Optimization of electricity production
3. Spotlight 3: Towards resilient, responsible decisions
4. A final (personal) word

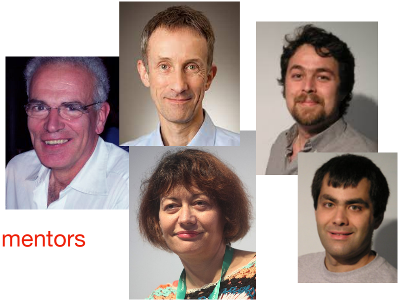
Emotional parenthesis...




mentors

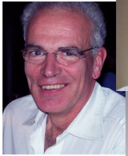
Emotional parenthesis...



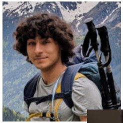
mentors

close colleagues

Emotional parenthesis...



mentors



PhDs

post-docs

close colleagues

Nonsmooth optimization at work: Outline

1. Spotlight 1: Do you know all about prox ?
2. Spotlight 2: Optimization of electricity production
3. Spotlight 3: Towards resilient, responsible decisions
4. A final (personal) word

Structured nonsmoothness: explicit case

For simple nonsmooth g, the nonsmoothness is explicit

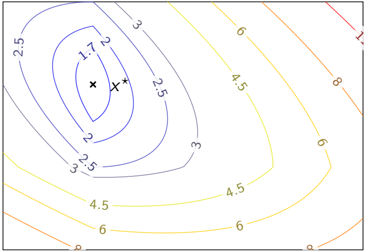
$$F(x) = f(x) + g(x)$$

$$F(x) = g \circ c(x)$$

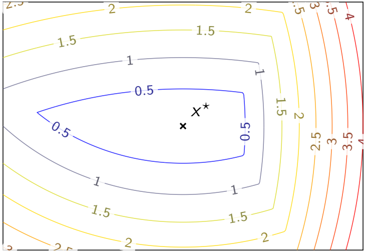
Examples: $g = \|\cdot\|_1$ and $g = \max$

Matrix examples: $g = \|\cdot\|_{\text{trace}}$ and $g = \lambda_{\max}$

$\min_x F(x) = \|Ax - b\|^2 + \lambda \|x\|_1$



$\min_x F(x) = \max(c_1, c_2, c_3)$



Structured nonsmoothness: explicit case

For simple nonsmooth g, the nonsmoothness is explicit

$$F(x) = f(x) + g(x)$$

$$F(x) = g \circ c(x)$$

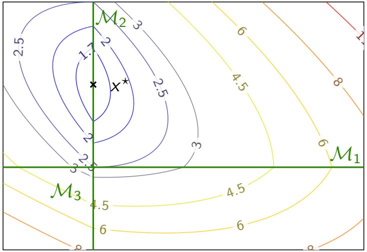
Examples: $g = \|\cdot\|_1$ and $g = \max$

Matrix examples: $g = \|\cdot\|_{\text{trace}}$ and $g = \lambda_{\max}$

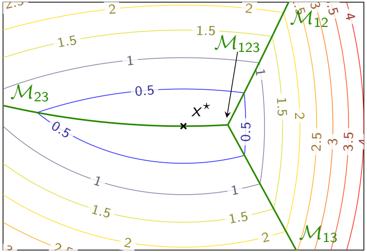
In many target applications, we observe that:

- nondiff. points organize in smooth manifolds $\mathcal{M}$
- locally, F is smooth along $\mathcal{M}$ and nonsmooth across $\mathcal{M}$

$\min_x F(x) = \|Ax - b\|^2 + \lambda\|x\|_1$



$\min_x F(x) = \max(c_1, c_2, c_3)$



Structured nonsmoothness: explicit case

For simple nonsmooth g, the nonsmoothness is explicit

$$F(x) = f(x) + g(x)$$

$$F(x) = g \circ c(x)$$

Examples: $g = \|\cdot\|_1$ and $g = \max$

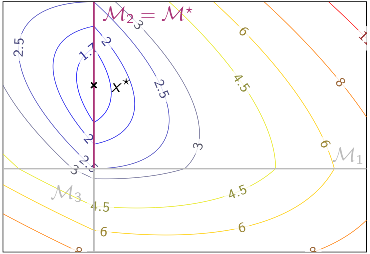
Matrix examples: $g = \|\cdot\|_{\mathrm{trace}}$ and $g = \lambda_{\max}$

In many target applications, we observe that:

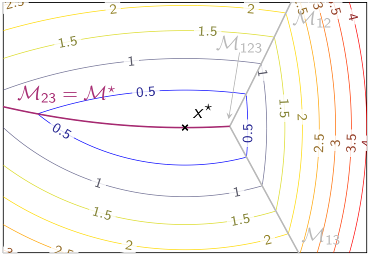
- nondiff. points organize in smooth manifolds $\mathcal{M}$
- locally, F is smooth along $\mathcal{M}$ and nonsmooth across $\mathcal{M}$
- there is an optimal manifold $\mathcal{M}^\star \ni x^\star$
- full first-order information ($\partial F(x)$ and more)

Can we detect $\mathcal{M}^\star$?

$\min_x F(x) = \|Ax - b\|^2 + \lambda\|x\|_1$



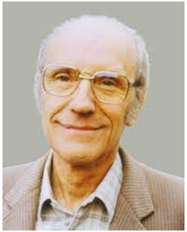
$\min_x F(x) = \max(c_1, c_2, c_3)$



Proximal operator: identification

J.J. Moreau, father of convex analysis, in the 1960s
("mécanique appliquée aux mathématiques")

Proximal operator $$\operatorname{prox}_{\gamma g}(y) = \underset{z}{\operatorname{argmin}} \left\{ g(z) + \frac{1}{2\gamma}\|z - y\|^2 \right\}$$

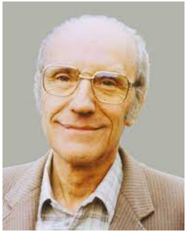


J.J. Moreau

Proximal operator: identification

J.J. Moreau, father of convex analysis, in the 1960s

("mécanique appliquée aux mathématiques")



J.J. Moreau

Proximal operator $$\operatorname{prox}_{\gamma g}(y) = \operatorname*{argmin}_{z} \left\{ g(z) + \frac{1}{2\gamma}\|z - y\|^2 \right\}$$

A. Daniilidis

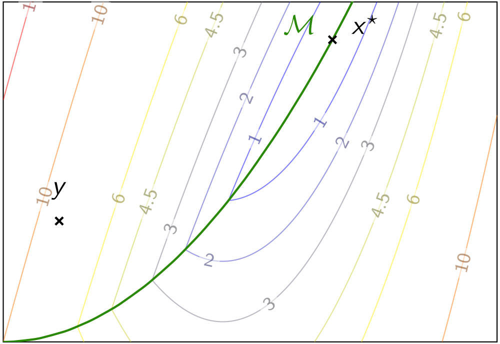
Gradient-proximal operator

(locally, smoothly) identifies $\mathcal{M}$

(under some natural assumptions)

[Daniilidis, Hare, Malick '06]

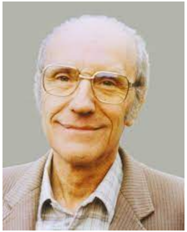
Grad-prox operator: $T(y) = \operatorname{prox}_{\gamma g}\left(y - \gamma \nabla f(y)\right)$



Proximal operator: identification

J.J. Moreau, father of convex analysis, in the 1960s

("mécanique appliquée aux mathématiques")



J.J. Moreau

Proximal operator $\quad \mathrm{prox}_{\gamma g}(y) = \operatorname*{argmin}_z \left\{ g(z) + \frac{1}{2\gamma}\|z - y\|^2 \right\}$

A. Daniilidis

Gradient-proximal operator

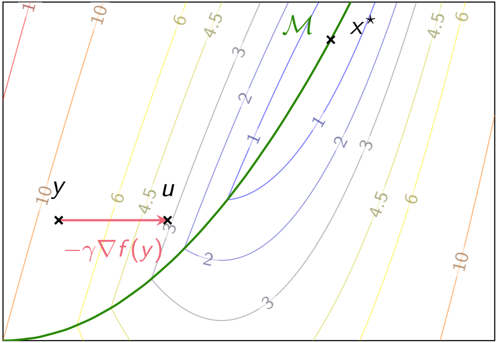
(locally, smoothly) identifies $\mathcal{M}$

(under some natural assumptions)

[Daniilidis, Hare, Malick '06]

Grad-prox operator: $\quad T(y) = \mathrm{prox}_{\gamma g}\left(y - \gamma \nabla f(y)\right)$

1. Explicit step on f: $\quad u = y - \gamma \nabla f(y)$

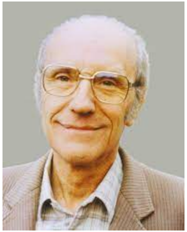


Proximal operator: identification

J.J. Moreau, father of convex analysis, in the 1960s

("mécanique appliquée aux mathématiques")

Proximal operator $\quad \mathrm{prox}_{\gamma g}(y) = \underset{z}{\mathrm{argmin}} \left\{ g(z) + \frac{1}{2\gamma}\|z-y\|^2 \right\}$



J.J. Moreau

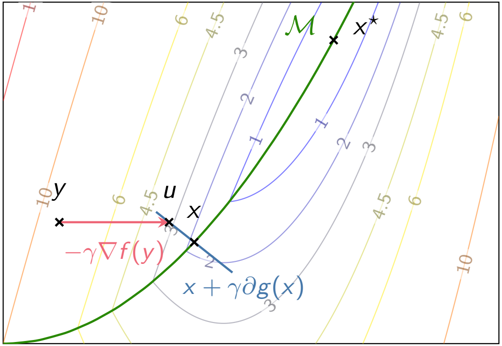
A. Daniilidis

Gradient-proximal operator

(locally, smoothly) identifies $\mathcal{M}$

(under some natural assumptions)

[Daniilidis, Hare, Malick '06]



Grad-prox operator: $\quad T(y) = \mathrm{prox}_{\gamma g}\left(y - \gamma\nabla f(y)\right)$

1. Explicit step on f: $\quad u = y - \gamma\nabla f(y)$
2. Implicit step on g:

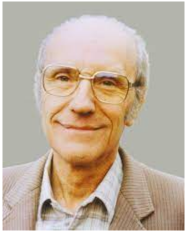
$$x = \mathrm{prox}_{\gamma g}(u) \;\Leftrightarrow\; u \in x + \gamma\partial g(x)$$

Proximal operator: identification

J.J. Moreau, father of convex analysis, in the 1960s

("mécanique appliquée aux mathématiques")

Proximal operator $\qquad \mathrm{prox}_{\gamma g}(y) = \operatorname*{argmin}_z \left\{ g(z) + \frac{1}{2\gamma}\|z-y\|^2 \right\}$



J.J. Moreau

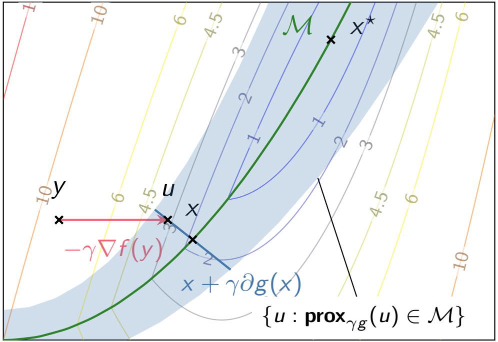
A. Daniilidis

Gradient-proximal operator

(locally, smoothly) identifies $\mathcal{M}$

(under some natural assumptions)

[Daniilidis, Hare, Malick '06]



Grad-prox operator: $\quad T(y) = \mathrm{prox}_{\gamma g}\left(y - \gamma\nabla f(y)\right)$

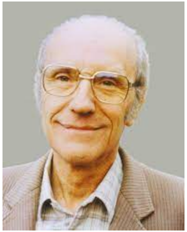
1. Explicit step on f: $\quad u = y - \gamma\nabla f(y)$
2. Implicit step on g:

$$x = \mathrm{prox}_{\gamma g}(u) \;\Leftrightarrow\; u \in x + \gamma\partial g(x)$$

Proximal operator: identification

J.J. Moreau, father of convex analysis, in the 1960s

("mécanique appliquée aux mathématiques")



J.J. Moreau

Proximal operator $\quad \operatorname{prox}_{\gamma g}(y) = \operatorname*{argmin}_z \left\{ g(z) + \frac{1}{2\gamma}\|z-y\|^2 \right\}$

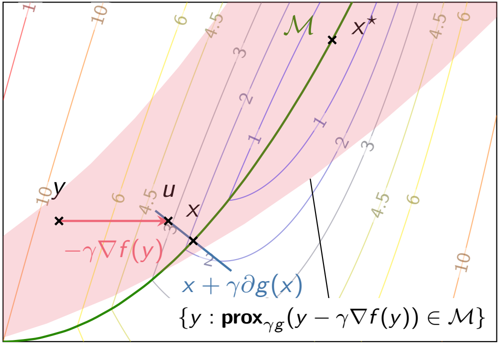
A. Daniilidis

Gradient-proximal operator

(locally, smoothly) identifies $\mathcal{M}$

(under some natural assumptions)

[Daniilidis, Hare, Malick '06]



Grad-prox operator: $\quad T(y) = \operatorname{prox}_{\gamma g}\left(y - \gamma\nabla f(y)\right)$

1. Explicit step on f: $\quad u = y - \gamma\nabla f(y)$
2. Implicit step on g:

$$x = \operatorname{prox}_{\gamma g}(u) \;\Leftrightarrow\; u \in x + \gamma\partial g(x)$$

How to exploit structure identification ?

Replace the nonsmooth problem $\min_{x \in \mathbb{R}^d} F(x)$ by the smooth problem $\min_{x \in \mathcal{M}^\star} F(x)$

Apply efficient 2nd order smooth (Riemannian) optimization algorithms...

Add constraints to simplify the problem

Simple idea [SMAI-MODE @ Le Havre '04], but not so simple in practice...

How to exploit structure identification ?

Replace the nonsmooth problem $\min_{x\in\mathbb{R}^d} F(x)$ by the smooth problem $\min_{x\in\mathcal{M}^\star} F(x)$

Apply efficient 2nd order smooth (Riemannian) optimization algorithms...

Add constraints to simplify the problem

Simple idea [SMAI-MODE @ Le Havre '04], but not so simple in practice...

Solution: Gilles Bareilles Ph.D. (2019-2022)

- interwine prox-grad steps and Newton-like steps
- guarantees on (global) convergence
- properly chosen parameters to identification and quadratic convergence
- "Newton acceleration of proximal-gradient method"

$+$ what happens in the case $g \circ c$!

geometry of the function vs. prox outputs

not in the same space

G. Bareilles

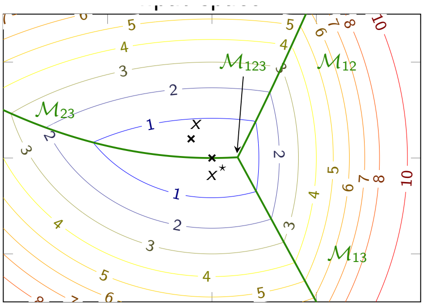
(2022 Dodu Prize)

Proximal identification for $F = g \circ c$

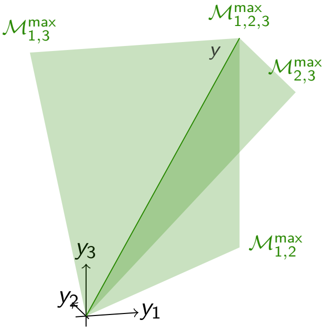
We have the prox of g... but not the prox of $F = g \circ c$

Still use $\text{prox}_{\gamma g}$, identify in the intermediate space, and then identify in the x-space

Ex: $F(x) = \max(c_1(x), c_2(x), c_3(x))$



$g(y) = \max(y_1, y_2, y_3)$



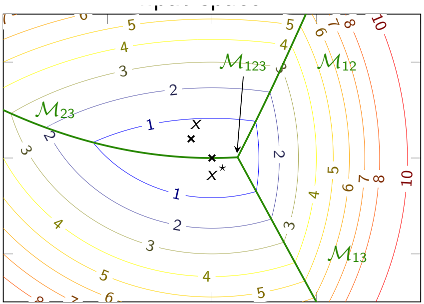
Proximal identification for $F = g \circ c$

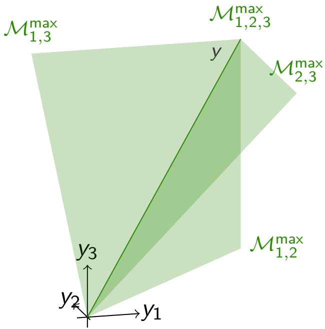
We have the prox of g... but not the prox of $F = g \circ c$

Still use $\mathrm{prox}_{\gamma g}$, identify in the intermediate space, and then identify in the x-space

Ex: $F(x) = \max(c_1(x), c_2(x), c_3(x))$

$g(y) = \max(y_1, y_2, y_3)$





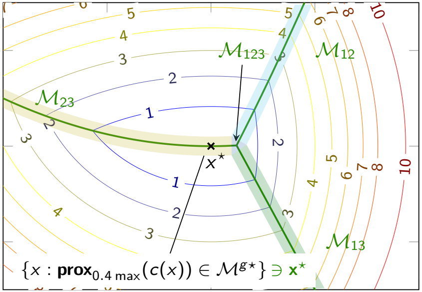
There is an "explicit" segment of stepsizes that gives identification [Bareilles Iutzeler Malick '22]

Proximal identification for $F = g \circ c$

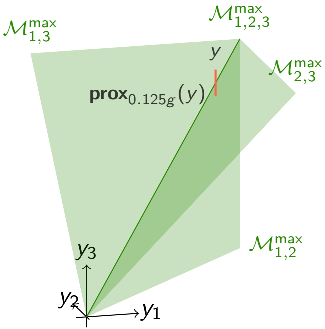
We have the prox of g... but not the prox of $F = g \circ c$

Still use $\text{prox}_{\gamma g}$, identify in the intermediate space, and then identify in the x-space

Ex: $F(x) = \max(c_1(x), c_2(x), c_3(x))$



$g(y) = \max(y_1, y_2, y_3)$



There is an "explicit" segment of stepsizes that gives identification [Bareilles Iutzeler Malick '22]

γ too small: detection of $\mathcal{M}^\star$ only near $x^\star$

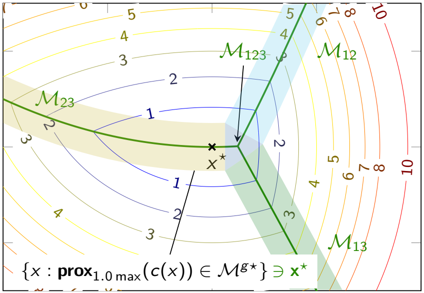
Proximal identification for $F = g \circ c$

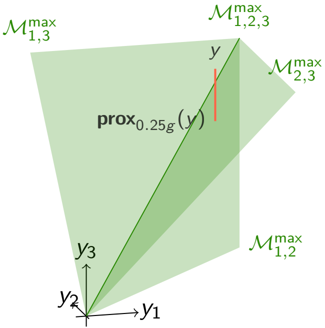
We have the prox of g… but not the prox of $F = g \circ c$

Still use $\mathrm{prox}_{\gamma g}$, identify in the intermediate space, and then identify in the x-space

Ex: $F(x) = \max(c_1(x), c_2(x), c_3(x))$

$g(y) = \max(y_1, y_2, y_3)$





There is an "explicit" segment of stepsizes that gives identification [Bareilles Iutzeler Malick '22]

γ too small: detection of $\mathcal{M}^\star$ only near $x^\star$

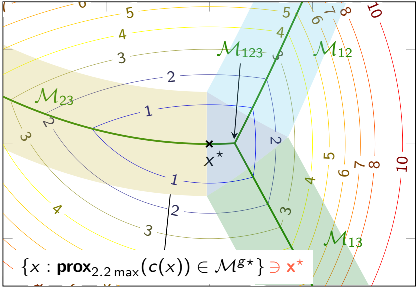
Proximal identification for $F = g \circ c$

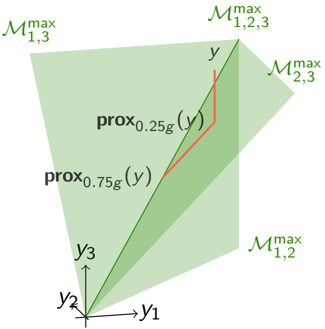
We have the prox of g... but not the prox of $F = g \circ c$

Still use $\mathrm{prox}_{\gamma g}$, identify in the intermediate space, and then identify in the x-space

Ex: $F(x) = \max(c_1(x), c_2(x), c_3(x))$

$g(y) = \max(y_1, y_2, y_3)$





There is an "explicit" segment of stepsizes that gives identification [Bareilles Iutzeler Malick '22]

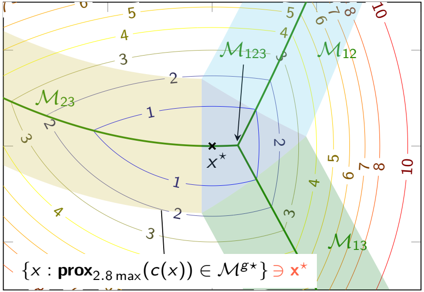
γ too small: detection of $\mathcal{M}^\star$ only near $x^\star$

Proximal identification for $F = g \circ c$

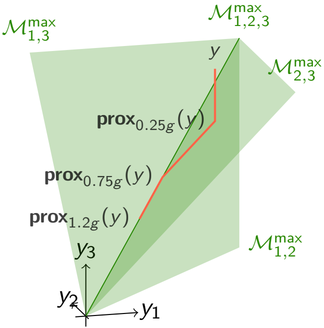
We have the prox of g... but not the prox of $F = g \circ c$

Still use $\mathrm{prox}_{\gamma g}$, identify in the intermediate space, and then identify in the x-space

Ex: $F(x) = \max(c_1(x), c_2(x), c_3(x))$



$g(y) = \max(y_1, y_2, y_3)$



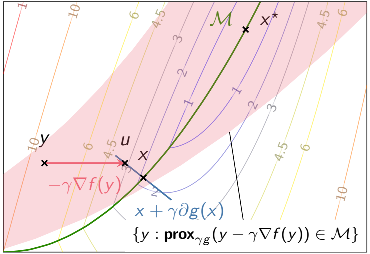
There is an "explicit" segment of stepsizes that gives identification [Bareilles Iutzeler Malick '22]

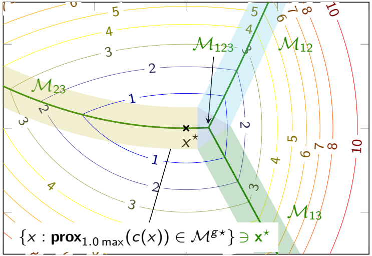
γ too small: detection of $\mathcal{M}^\star$ only near $x^\star$ γ too big: no more detection of $\mathcal{M}^\star$ near $x^\star$

So we can properly interlace Newton-like steps ☺

Conclusion on this spotlight

- Nonsmoothness is highly structured
- Sometimes, we know "explicitly" the structure (thank you, prox)
- We can exploit it: Newton acceleration ($\neq$ Nesterov acceleration)
- Applications on matrix problems E.g. $F(x) = \lambda_{\max}\left(A_0 + \sum_{i=1}^{n} x_i A_i\right)$





Nonsmooth optimization at work: Outline

1. Spotlight 1: Do you know all about prox ?
2. Spotlight 2: Optimization of electricity production
3. Spotlight 3: Towards resilient, responsible decisions
4. A final (personal) word

Finding "optimal" production schedules

In France: EDF produces electricity by N production units

nuclear 63% renewables 14% oil/gaz/coal 12% hydro 17%




Day-to-day optimization of production "unit-commitment" (compute a minimal-cost production schedule, satisfying operational constraints and meeting customer demand, over T times).

Hard optimization problem: large-scale, heterogeneous, complex ($\geqslant 10^6$ variables, $\geqslant 10^6$ constraints)

$$\binom{\text{simplified}}{\text{model}} \quad \left\{ \begin{array}{ll} \min \quad \sum_i c_i^\top x_i & \text{(production costs)} \\ \quad \sum_i x_i = d & \text{(demand constraints)} \\ \quad (x_1, \ldots, x_N) \in X_1 \times \cdots \times X_N & \text{(operational constraints)} \end{array} \right.$$

Out of reach for (mixed-integer linear) solvers... But where is the nonsmoothness ?

Finding "optimal" production schedules

In France: EDF produces electricity by N production units

nuclear 63%

renewables 14%

oil/gaz/coal 12%

hydro 17%

Day-to-day optimization of production "unit-commitment" (compute a minimal-cost production schedule, satisfying operational constraints and meeting customer demand, over T times).

Hard optimization problem: large-scale, heterogeneous, complex ($\geqslant 10^6$ variables, $\geqslant 10^6$ constraints)

$$\begin{pmatrix}\text{simplified}\\ \text{model}\end{pmatrix} \quad \left\{ \begin{array}{lll} \min & \sum_i c_i^\top x_i & \text{(production costs)} \\ & \sum_i x_i = d \quad \leftarrow u \in \mathbb{R}^T & \text{(demand constraints)} \\ & (x_1, \ldots, x_N) \in X_1 \times \cdots \times X_N & \text{(operational constraints)} \end{array} \right.$$

Out of reach for (mixed-integer linear) solvers... But where is the nonsmoothness ?

Lagrangian decomposition

- Dual function (concave)

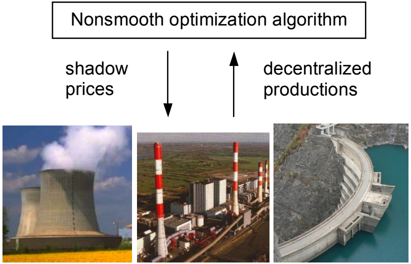
$$\theta(u) = \left\{ \begin{array}{l} \min \quad \sum_{i=1}^{N} c_i^\top x_i + \sum_{t=1}^{T} u^t \left(d^t - \sum_{i=1}^{N} x_i^t \right) \\ \quad (x_1, \ldots, x_N) \in X_1 \times \cdots \times X_N \end{array} \right.$$

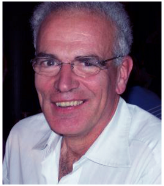
- Dualizing the coupling constraint makes it decomposable by units

$$\theta(u) = d^\top u + \sum_{i=1}^{N} \theta_i(u)$$

$$\theta_i(u) = \left\{ \begin{array}{l} \min \quad (c_i - u)^\top x_i \\ \quad x_i \in X_i \end{array} \right.$$

- Nonsmooth algorithm: inexact prox. bundle [Lemaréchal '75... '95]




C. Lemarechal


S. Charousset

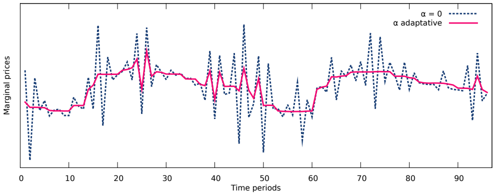

A. Renaud

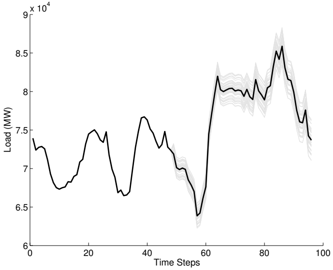
- Research in the 1990's
- Production in early 2000's
- Save money and CO2 !

On the shoulders of giants

Our work

- Denoising dual solutions (by TV-regularization) [Zaourar, Malick '13]
- Acceleration of the bundle method (using coarse linearizations) [Malick, Oliveira, Zaourar '15]
- (Level) asynchronous bundle algorithm [Iutzeler, Malick, Oliveira '18]
- Introducing weather uncertainty in the model
  - robust version of the problem + bundle method [van Ackooij, Lebbe, Malick '16]
  - 2-stage stochastic version + double decomposition algorithm [van Ackooij, Malick '15]

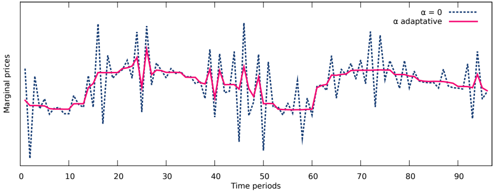


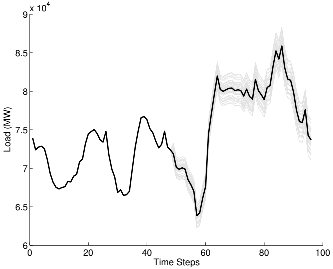


On the shoulders of giants

Our work

- Denoising dual solutions (by TV-regularization) [Zaourar, Malick '13]
- Acceleration of the bundle method (using coarse linearizations) [Malick, Oliveira, Zaourar '15]
- (Level) asynchronous bundle algorithm [Iutzeler, Malick, Oliveira '18]
- Introducing weather uncertainty in the model
  - robust version of the problem + bundle method [van Ackooij, Lebbe, Malick '16]
  - 2-stage stochastic version + double decomposition algorithm [van Ackooij, Malick '15]





Two-stage stochastic unit-commitment

W. van Ackooij

- The schedule x is sent to the grid-operator (RTE) before being activated and before observing uncertainty
- In real time, a new production schedule can be sent at certain times
- At time τ, we have the observed load $\xi_1, ..., \xi_\tau$ and the current best forecast $\xi_{\tau+1}, ..., \xi_T$
- We propose a stochastic 2-stage problem:

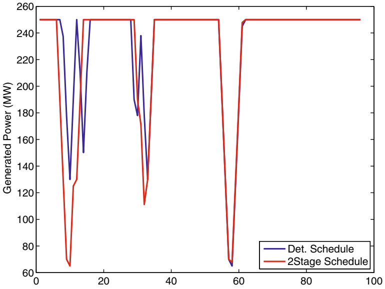
$$\boxed{\left\{ \begin{array}{ll} \min & c^\top x + \mathbb{E}[c(x,\xi)] \\ & x \in X, \quad \sum_i x_i = d \end{array} \right.} \qquad \text{where } c(x,\xi) = \left\{ \begin{array}{ll} \min & c^\top y \\ & y \in X, \quad \sum_i y_i = \xi \\ & y \text{ coincides with } x \text{ on } 1, \ldots, \tau \end{array} \right.$$

  - 2nd stage model: same as 1st stage but with smaller horizon
  - fine operational modeling vs difficult to compute
  - complexity of $c(x,\xi)$ only allows for simple modeling of randomness

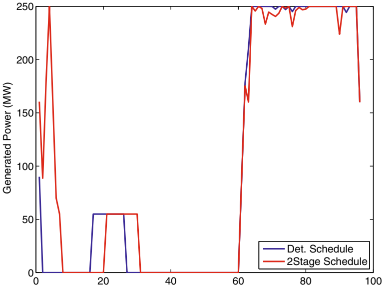
- New algo: double decomposition (by units and scenarios) using the same ingredients

Numerical illustration for stochastic unit-commitment

- On a 2013 EDF instance (medium-size)
  - deterministic problem : 50k continuous variables, 27k binary variables, 815k constraints
  - stochastic version (50 scenarios) : 1,200k continuous var., 700k binary var., 20,000k constraints
- Our method allows to solve it ☺ (in reasonable time)
- Observation: generation transferred from cheap/inflexible to expensive/flexible
- Example: production schedules for 2 units: determinist vs stochastic



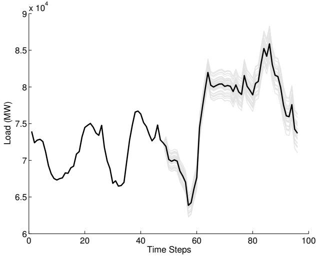
cheap/inflexible unit (nuclear)



expensive/flexible unit (gaz)

Conclusion on this spotlight

- Electricity managment optimzation is huge
- Ad: attend Sandrine's talk this afternoon for a broader view
- Nonsmoothness 1: Lagrangian decomposition
- Nonsmoothness 2: robustness against (weather) uncertainties



Nonsmooth optimization at work: Outline

1. Spotlight 1: Do you know all about prox ?
2. Spotlight 2: Optimization of electricity production
3. Spotlight 3: Towards resilient, responsible decisions
4. A final (personal) word

Deep learning can be impressive

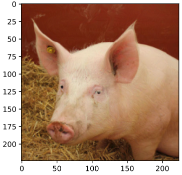
Spectacular success of deep learning, in many fields/applications... E.g. in generation

Ex: picture generated with stable diffusion (`https://stablediffusionweb.com`)

"towards resilient, responsible decisions"

Example #1: Don't forget how fragile deep learning can be !

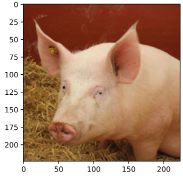
Illustration 1: Flying pigs (notebooks of NeurIPS 2018, tutorial on robustness)



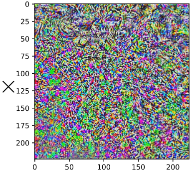
pig (99%)

Example #1: Don't forget how fragile deep learning can be !

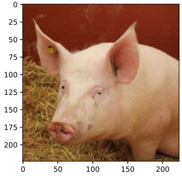
Illustration 1: Flying pigs (notebooks of NeurIPS 2018, tutorial on robustness)



$+\ \mathcal{E}\times$



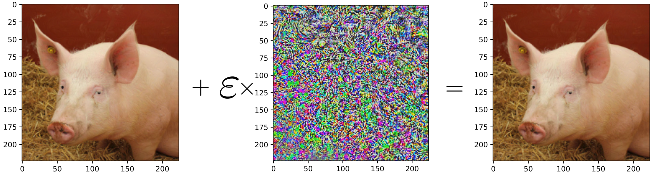
$=$



pig (99%)

Example #1: Don't forget how fragile deep learning can be !

Illustration 1: Flying pigs (notebooks of NeurIPS 2018, tutorial on robustness)

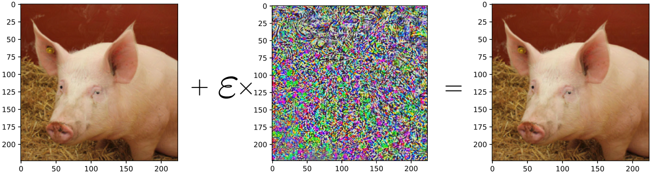


pig (99%) airliner (96%)

"ML is a wonderful technology: it makes pigs fly" [Kolter, Madry '18]

Example $\#1$: Don't forget how fragile deep learning can be !

Illustration 1: Flying pigs (notebooks of NeurIPS 2018, tutorial on robustness)



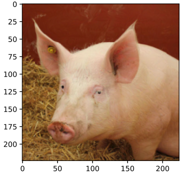
pig (99%) airliner (96%)

"ML is a wonderful technology: it makes pigs fly" [Kolter, Madry '18]

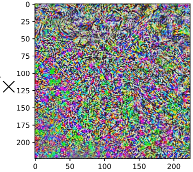
Illustration 2: Attacks against self-driving cars [@ CVPR '18]

Example $\#1$: Don't forget how fragile deep learning can be !

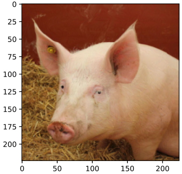
Illustration 1: Flying pigs (notebooks of NeurIPS 2018, tutorial on robustness)



$+\ \mathcal{E} \times$



$=$



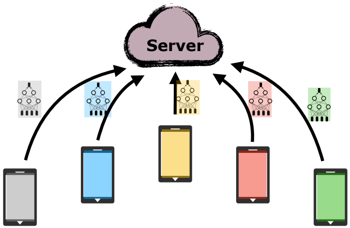
pig (99%) airliner (96%)

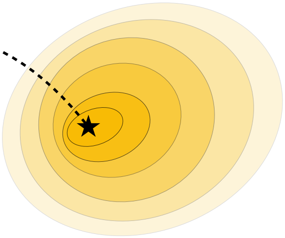
"ML is a wonderful technology: it makes pigs fly" [Kolter, Madry '18]

Illustration 2: Attacks against self-driving cars [@ ICLR '19]

Example #2: ML may perform poorly for some people

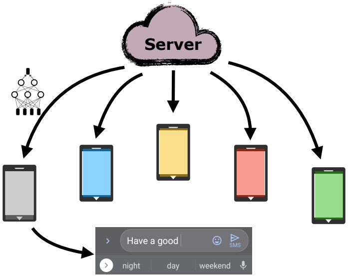
Example: Global model is trained on *average distribution* across clients (ERM)

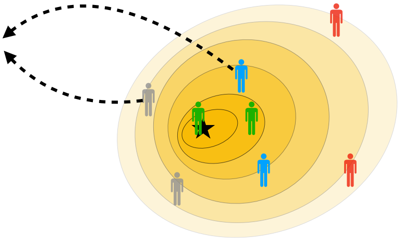




Example #2: ML may perform poorly for some people

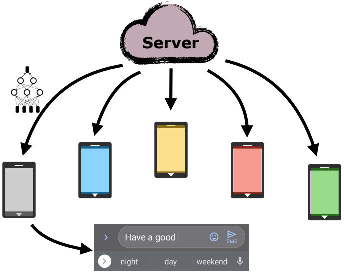
Example: Global model is deployed on *individual* clients

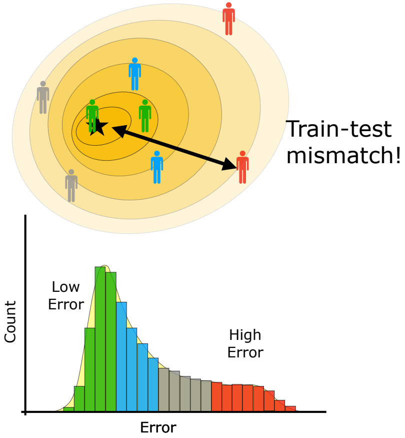




Example #2: ML may perform poorly for some people

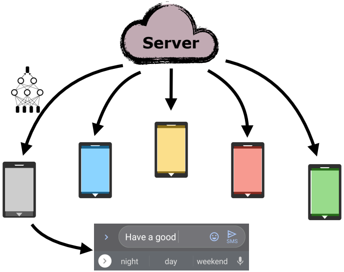
Example: Global model is deployed on *individual* clients

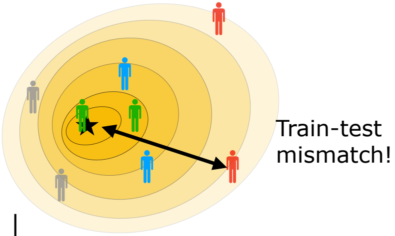


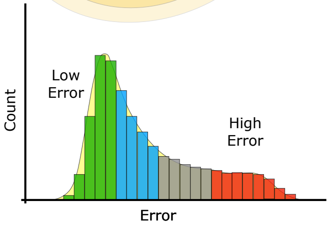


Example #2: ML may perform poorly for some people

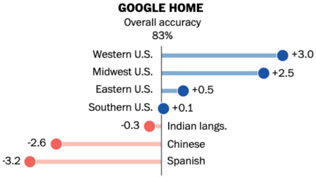
Example: Global model is deployed on *individual* clients

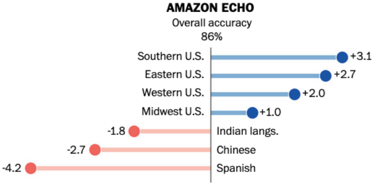






From Washington Post (2019) “the accent gap”





Optimization set-up

- Training data: $\xi_1, \ldots, \xi_N$

  e.g. in supervised learning: labeled data $\xi_i = (a_i, y_i)$ feature, label

- Train model: $f(x, \cdot)$ the loss function with x the parameter/decision $(\omega, \beta, \theta, \ldots)$

  e.g. least-square regression: $f\big(x, (a, y)\big) = (x^\top a - y)^2$

- Compute x via empirical risk minimization (a.k.a SAA)

$$\min_x \; \frac{1}{N} \sum_{i=1}^{N} f(x, \xi_i) = \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x, \xi)] \qquad \text{with } \widehat{\mathbb{P}}_N = \frac{1}{N} \sum_{i=1}^{N} \delta_{\xi_i}$$

Optimization set-up

- Training data: $\xi_1, \ldots, \xi_N$

  e.g. in supervised learning: labeled data $\xi_i = (a_i, y_i)$ feature, label

- Train model: $f(x, \cdot)$ the loss function with x the parameter/decision $(\omega, \beta, \theta, \ldots)$

  e.g. least-square regression: $f\big(x, (a, y)\big) = (x^\top a - y)^2$

- Compute x via empirical risk minimization (a.k.a SAA)

$$\min_x \frac{1}{N} \sum_{i=1}^{N} f(x, \xi_i) = \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x, \xi)] \qquad \text{with } \widehat{\mathbb{P}}_N = \frac{1}{N} \sum_{i=1}^{N} \delta_{\xi_i}$$

- Prediction with x for different data ξ
  - Adversarial attacks (e.g. flying pigs, driving cakes...)
  - Presence of bias, e.g. heterogeneous data
  - Distributional shifts: $\mathbb{P}_{\text{train}} \neq \mathbb{P}_{\text{test}}$

- Solution: take possible variations into account during training

...and nonsmoothness comes into play ☺

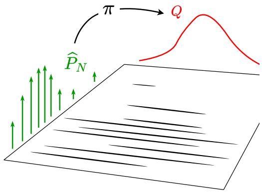
(Wasserstein) Distributionally Robust Optimization

Rather than $\min_{x} \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x,\xi)]$ solve instead $\min_{x} \max_{\mathbb{Q}\in\mathcal{U}} \mathbb{E}_{\mathbb{Q}}[f(x,\xi)]$

with $\mathcal{U}$ a neighborhood of $\widehat{\mathbb{P}}_N$

(Wasserstein) Distributionally Robust Optimization

Rather than $\min_{x} \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x,\xi)]$ solve instead $\min_{x} \max_{\mathbb{Q}\in\mathcal{U}} \mathbb{E}_{\mathbb{Q}}[f(x,\xi)]$



with $\mathcal{U}$ a neighborhood of $\widehat{\mathbb{P}}_N$

Wasserstein balls as ambiguity sets

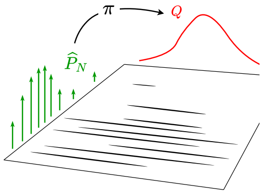
$$\mathcal{U} = \{\ \mathbb{Q} : W(\widehat{\mathbb{P}}_N, \mathbb{Q}) \leqslant \rho\ \}$$

$$W(\widehat{\mathbb{P}}_N, \mathbb{Q}) = \min_{\boldsymbol{\pi}} \left\{ \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] : [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \right\}$$

(Wasserstein) Distributionally Robust Optimization

Rather than $\min_{x} \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x,\xi)]$ solve instead $\min_{x} \max_{\mathbb{Q}\in\mathcal{U}} \mathbb{E}_{\mathbb{Q}}[f(x,\xi)]$

with $\mathcal{U}$ a neighborhood of $\widehat{\mathbb{P}}_N$



Wasserstein balls as ambiguity sets

$$\mathcal{U} = \{ \mathbb{Q} : W(\widehat{\mathbb{P}}_N, \mathbb{Q}) \leqslant \rho \}$$

$$W(\widehat{\mathbb{P}}_N, \mathbb{Q}) = \min_{\boldsymbol{\pi}} \left\{ \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] : [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \right\}$$

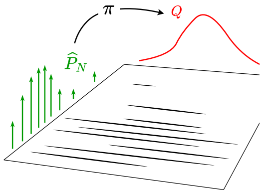
WDRO objective function for given x, $\widehat{\mathbb{P}}_N$, ρ

$$\left\{ \begin{array}{c} \max_{\mathbb{Q}} \ \mathbb{E}_{\mathbb{Q}}[f(x,\xi)] \\ W(\widehat{\mathbb{P}}_N, \mathbb{Q}) \leqslant \rho \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} \max_{\mathbb{Q},\boldsymbol{\pi}} \ \mathbb{E}_{\mathbb{Q}}[f(x,\xi)] \\ \ [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \\ \ \min_{\boldsymbol{\pi}} \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] \leqslant \rho \end{array} \right.$$

(Wasserstein) Distributionally Robust Optimization

Rather than $\min_{x} \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x,\xi)]$ solve instead $\min_{x} \max_{\mathbb{Q}\in\mathcal{U}} \mathbb{E}_{\mathbb{Q}}[f(x,\xi)]$

with $\mathcal{U}$ a neighborhood of $\widehat{\mathbb{P}}_N$



Wasserstein balls as ambiguity sets

$$\mathcal{U} = \{\ \mathbb{Q} : W(\widehat{\mathbb{P}}_N, \mathbb{Q}) \leqslant \rho\ \}$$

$$W(\widehat{\mathbb{P}}_N, \mathbb{Q}) = \min_{\boldsymbol{\pi}} \Big\{ \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] : [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \Big\}$$

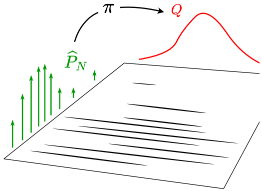
WDRO objective function for given x, $\widehat{\mathbb{P}}_N$, ρ

$$\left\{\begin{array}{c} \max_{\mathbb{Q}}\ \mathbb{E}_{\mathbb{Q}}[f(x,\xi)] \\ W(\widehat{\mathbb{P}}_N, \mathbb{Q}) \leqslant \rho \end{array}\right. \Leftrightarrow \left\{\begin{array}{l} \max_{\mathbb{Q},\boldsymbol{\pi}}\ \mathbb{E}_{\mathbb{Q}}[f(x,\xi)] \\ \ [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \\ \ \min_{\boldsymbol{\pi}} \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] \leqslant \rho \end{array}\right. \Leftrightarrow \left\{\begin{array}{l} \max_{\boldsymbol{\pi}}\ \mathbb{E}_{[\boldsymbol{\pi}]_2}[f(x,\xi)] \\ \ [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N \\ \ \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] \leqslant \rho \end{array}\right.$$

(Wasserstein) Distributionally Robust Optimization

Rather than $\min\limits_{x} \mathbb{E}_{\widehat{\mathbb{P}}_N}[f(x,\xi)]$ solve instead $\min\limits_{x} \max\limits_{\mathbb{Q}\in\mathcal{U}} \mathbb{E}_{\mathbb{Q}}[f(x,\xi)]$

with $\mathcal{U}$ a neighborhood of $\widehat{\mathbb{P}}_N$



Wasserstein balls as ambiguity sets

$$\mathcal{U} = \{\, \mathbb{Q} : W(\widehat{\mathbb{P}}_N, \mathbb{Q}) \leqslant \rho \,\}$$

$$W(\widehat{\mathbb{P}}_N, \mathbb{Q}) = \min_{\boldsymbol{\pi}} \Big\{ \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] : [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \Big\}$$

WDRO objective function for given x, $\widehat{\mathbb{P}}_N$, ρ

$$\left\{\begin{array}{l} \max_{\mathbb{Q}}\ \mathbb{E}_{\mathbb{Q}}[f(x,\xi)] \\ W(\widehat{\mathbb{P}}_N,\mathbb{Q}) \leqslant \rho \end{array}\right. \Leftrightarrow \left\{\begin{array}{l} \max_{\mathbb{Q},\boldsymbol{\pi}}\ \mathbb{E}_{\mathbb{Q}}[f(x,\xi)] \\ [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N, [\boldsymbol{\pi}]_2 = \mathbb{Q} \\ \min_{\boldsymbol{\pi}} \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] \leqslant \rho \end{array}\right. \Leftrightarrow \left\{\begin{array}{l} \max_{\boldsymbol{\pi}}\ \mathbb{E}_{[\boldsymbol{\pi}]_2}[f(x,\xi)] \\ [\boldsymbol{\pi}]_1 = \widehat{\mathbb{P}}_N \\ \mathbb{E}_{\boldsymbol{\pi}}[c(\xi,\xi')] \leqslant \rho \end{array}\right.$$

$$\Leftrightarrow \min_{\lambda\geqslant 0} \lambda\rho + \mathbb{E}_{\widehat{\mathbb{P}}_N}\big[\max_{\xi'} \{f(x,\xi') - \lambda c(\xi,\xi')\}\big]$$

...(finite dimension) nonsmooth... great talk of Tam Le yesterday ☺

...computable in some (specific) cases [Kuhn *et al.* '18]

...actually many more since the PhD of Florian Vincent, see poster tomorow ☺

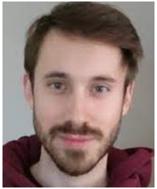
Current research in my group

Our work

- Toolbox: robustify our model with `skWDRO` [Vincent, Azizian, Iutzeler, Malick '24]

  scikitlearn interface + pytorch wrapper

- Generalization guarantees [Le, Malick '24] [Azizian, Iutzeler, Malick '23]
- (abstract, entropic) regularizations of WDRO [Azizian, Iutzeler, Malick '22]
- Applications in federated learning [Laguel, Pillutla, Harchaoui, Malick '23]



F. Iutzeler

W. Azizian

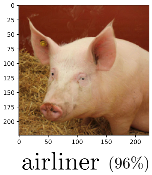
Y. Laguel

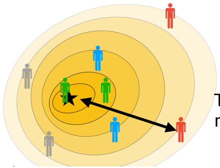
Tam Le

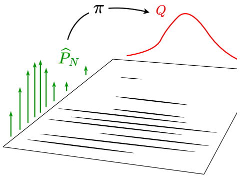
F. Vincent

Conclusion on this spotlight

- Deep learning works very well... unless it does not.
- Need for more robustness (resilience, fairness...) – brought by max/nonsmoothness
- Wasserstein DRO is a nice playground – current work of my group
- Ad: Go and see Florian's poster... and robustify your models !







Nonsmooth optimization at work: Outline

1. Spotlight 1: Do you know all about prox ?
2. Spotlight 2: Optimization of electricity production
3. Spotlight 3: Towards resilient, responsible decisions
4. **A final (personal) word**

Back to the future

12ème journées du groupe MODE

$\mathcal{U}$-Lagrangien et géométrie

Jérôme MALICK[1], Scott MILLER[2]

[1] INRIA (Rhône-Alpes)
Montbonnot, 38334 St Ismier
jerome.malick@inria.fr

[2] University of California, San Diego
9500 Gilman Dr, m/c 0411, La Jolla, CA 92093-0411
scott@turbulence.ucsd.edu

RESUME

La méthode de Newton peut être considérée comme le prototype des algorithmes rapides d'optimisation. Dans cet exposé, nous comparons différentes manières de l'étendre à des problèmes d'optimisation non lisse. Les précisions sur le contenu de l'exposé se trouvent dans [3].

Le cadre de travail est le suivant. On s'intéresse à la minimisation sur $\mathbb{R}^n$ d'une fonction convexe f, et on suppose que le minimum est atteint sur une sous-variété $\mathcal{M}$ par apport à laquelle f est partly-smooth. Introduite dans [2], la partial smoothness exprime essentiellement que la régularité de f est confinée à $\mathcal{M}$. Le problème se reformule comme un problème de minimisation sous contraintes

$$\begin{cases} \min f(x) \\ x \in \mathcal{M}\,. \end{cases}$$

L'objectif est de préciser les liens entre différentes manières adapter la méthode de Newton à ce problème:

– les algorithmes provenant de la théorie du $\mathcal{U}$-Lagrangian de [1],
– les méthodes SQP,
– les méthodes de Newton locales sur $\mathcal{M}$.

Mots-clé: optimisation non lisse, partial smoothness, géométrie riemannienne

Classification AMS: 49J52, 65K10, 58C99

Références

[1] C. Lemaréchal, F. Oustry, and C. Sagastizábal : The $\mathcal{U}$-Lagrangian of a convex function. *Trans. AMS*, 352(2):711–729 (1999).

[2] A. S. Lewis : Active sets, nonsmoothness and sensitivity. *SIAM J. Optimization*, 13:702–725 (2003) .

[3] S. Miller, J. Malick : Connections between $\mathcal{U}$-Lagrangian, Riemannian Newton and SQP Methods for Convex Minimization. (2004, submitted for publication).

- From Le Havre to Lyon, nonsmoothness matters
- From 2004 to 2024, what a journey !
- Optimisation rules !
- CNRS/Insis topic of the year 2024
  (save the date: Oct.3 @ Paris)
- Theory $\longleftrightarrow$ Practice
- Optim $\longleftrightarrow$ ML
  (e.g. talk of Emilie Chouzenoux yesterday)
- Responsible decision-making

Many thanks !

Merci à vous

pour votre attention aujourd'hui

et pour faire vivre notre communauté demain – rdv en 2044 ?!

Et merci à eux