IASD App
2025-2026

OPTIMISATION POUR L'APPRENTISSAGE AUTOMATIQUE

3 février 2026 (Après-midi)

Cette séance : Régularisation
Algorithmes proximaux
Quelques exemples

A venir : Feuille d'exercices mise à jour

MÉTHODES PROXIMALES ET RÉGULARISATION

Motivation :

$$(P) \quad \underset{w \in \mathbb{R}^d}{\text{minimize}} \underbrace{\frac{1}{n} \sum_{i=1}^{n} f_i(w)}_{f(w)}$$

f_i dépend de données

- Si le problème possède plusieurs solutions, laquelle choisir ?
- Si on a un a priori sur la solution, comment faire en sorte que la solution de (P) ait ces propriétés ?

Ex) (X, y) jeu de données pour la régression

$$X = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

avec $y = Xw^* + \varepsilon$

ε : bruit

w^* : vraie tendance linéaire

w^* possède beaucoup de coefficients nuls (99%) ← [w^* est "Sparse" (ou creux, ou parcimonieux)

But : Modifier le problème de départ pour prendre en compte des propriétés (réelles ou souhaitées) sur la solution

Définition : Problème régularisé

$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \quad f(w) + \lambda r(w)$$

- $f : \mathbb{R}^d \to \mathbb{R}$ terme d'attache aux données

$$f(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w)$$

- $r : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ terme de régularisation (a priori indépendant des données)

- $\lambda \geq 0$: Paramètre de régularisation

↳ Lorsque $\lambda = 0$, le problème est le problème d'origine

$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \; f(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w)$$

↳ Lorsque $\lambda \to \infty$, le problème est équivalent à

$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \; r(w)$$ (plus d'impact des données)

↳ Le problème avec $\lambda > 0$ correspond à pondérer l'effet de la régularisation par rapport à l'attache aux données

(1) Régularisation ℓ_2 ("ridge" / écrêtée, Tychonov, ...)

$$r(w) = \frac{1}{2} \|w\|^2 = \frac{1}{2} \sum_{j=1}^{d} w_j^2$$

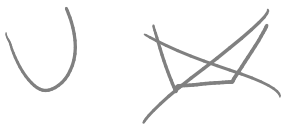
$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \; f(w) + \frac{\lambda}{2} \|w\|^2$$

But : → Réduire la sensibilité de la solution vis-à-vis des données (générique)

→ Convexifier le problème (spécifique à l'optimisation)

- $w \mapsto \frac{\lambda}{2}\|w\|^2$ est λ-fortement convexe

(φ λ-fortement convexe
$\Leftrightarrow \varphi(\alpha v + (1-\alpha)w) \leq \alpha\varphi(v) + (1-\alpha)\varphi(w) - \frac{\lambda}{2}\alpha(1-\alpha)\|v-w\|^2$)



- Si f est convexe, alors $f + \frac{\lambda}{2}\|\cdot\|^2$ est λ-fortement convexe

(unique minimum global)

Illustration

$$\underbrace{\min_{w\in\mathbb{R}^2} \frac{1}{4}\|Xw - y\|^2 + \frac{\lambda}{2}\|w\|^2}_{f_\lambda(w)} \qquad \lambda \geq 0$$

$$X = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \qquad y = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$\lambda = 0$: $\operatorname{argmin}_w f_0(w) = \left\{ \begin{bmatrix} 1 \\ w_2 \end{bmatrix} \mid w_2 \in \mathbb{R} \right\}$

$$f_0(w) = \frac{1}{4}\left((w_1 - 1)^2 + 1\right)$$

- Supposons qu'on perturbe $X \to X_\varepsilon = \begin{bmatrix} 1 & 0 \\ 0 & \varepsilon \end{bmatrix}$ avec $0 < \varepsilon \ll 1$

Alors $\operatorname{argmin} \frac{1}{4}\|X_\varepsilon w - y\|^2 = \left\{ \begin{bmatrix} 1 \\ 1/\varepsilon \end{bmatrix} \right\}$

$$\left\| \begin{bmatrix} 1 \\ 1/\varepsilon \end{bmatrix} \right\| = O(\varepsilon^{-1})$$

$\Rightarrow$ le problème est sensible à cette perturbation des données

$\lambda > 0$:

minimiser $f_\lambda(w) = \frac{1}{4}\|Xw - y\|^2 + \frac{\lambda}{2}\|w\|^2$

f_λ λ-fortement convexe

$$\nabla f_\lambda(w) = \frac{1}{2} X^T(Xw - y) + \lambda w$$

$$\nabla f_\lambda(w) = 0 \iff X^T X w - X^T y + 2\lambda w = 0$$

$$\iff (X^T X + 2\lambda I) w = X^T y$$

$X = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ $\quad X^T y = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$X^T X = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$

$$\iff \begin{bmatrix} 1+2\lambda & 0 \\ 0 & 2\lambda \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\iff w = \begin{bmatrix} \frac{1}{1+2\lambda} \\ 0 \end{bmatrix}$$

$$\operatorname{argmin} f_\lambda(w) = \left\{ \begin{bmatrix} \frac{1}{1+2\lambda} \\ 0 \end{bmatrix} \right\}$$

$\left\| \begin{bmatrix} \frac{1}{1+2\lambda} \\ 0 \end{bmatrix} \right\| = O(\frac{1}{\lambda})$

- Unique solution (contrairement au cas $\lambda = 0$)
- Quand $\lambda \to \infty$

$$\begin{bmatrix} \frac{1}{1+2\lambda} \\ 0 \end{bmatrix} \to \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$\left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\} = \operatorname{argmin}_w \frac{1}{2}\|w\|^2$

Pour le problème perturbé,

$$\operatorname{argmin} \frac{1}{4}\|X_\varepsilon w - y\|^2 + \frac{\lambda}{2}\|w\|^2 = \left\{ \begin{bmatrix} \frac{1}{1+2\lambda} \\ \frac{\varepsilon}{\varepsilon^2 + 2\lambda} \end{bmatrix} \right\}$$

$$\left(X_\varepsilon^T X_\varepsilon + 2\lambda I\right)\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = X_\varepsilon^T y$$

$$\begin{bmatrix} 1 & 0 \\ 0 & \varepsilon^2 \end{bmatrix} + \begin{bmatrix} 2\lambda & 0 \\ 0 & 2\lambda \end{bmatrix} \qquad \begin{bmatrix} 1 \\ \varepsilon \end{bmatrix}$$

La norme de la solution régularisée est d'ordre

$O(\varepsilon)$ pour ε petit, donc $\to 0$ $\varepsilon \to 0$

$\Rightarrow$ le problème avec régularisation est moins sensible aux perturbations dans les données

↳ Algorithmiquement, la régularisation ℓ_2 est facile à intégrer dans une méthode de type gradient (descente de gradient, gradient stochastique, voire sous-gradient)

Ex) Si f est C^1, alors

$$w_{k+1} = w_k - \alpha_k \nabla f(w_k)$$

Descente de gradient sur le problème sans régularisation ($\lambda = 0$)

$$w_{k+1} = w_k - \alpha_k \left(\nabla f(w_k) + \lambda w_k\right)$$

$$= \left(1 - \lambda\alpha_k\right) w_k - \alpha_k \nabla f(w_k)$$

Descente de régularisation pour le problème régularisé $\lambda > 0$

$1 - \lambda\alpha_k \in (0,1)$ pour α_k suffisamment petit

↳ Réduction des composantes de w_k par un facteur multiplicatif à chaque itération

"Weight decay"

NB: . SGD de Pytorch / Descente de gradient avec "weight decay"

⇒ ——————— sur problème avec régularisation ℓ_2

* SGD avec momentum

. Adam / Tuan ⑤ ont aussi un paramètre de weight decay, mais qui n'est pas équivalent à une régularisation ℓ_2 à cause du momentum ⇒ A l'origine de AdamW

② Régularisation ℓ_1 (LASSO, ...)

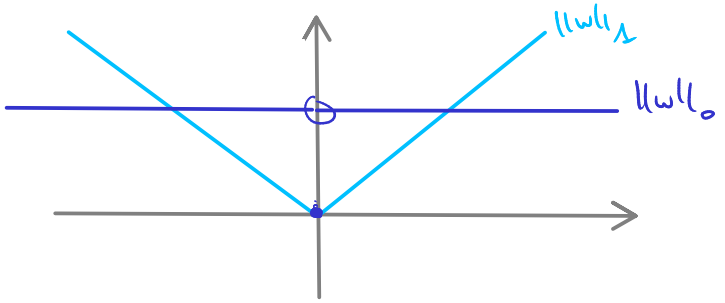
$$r(w) = \|w\|_1 = \sum_{j=1}^{d} |w_j|$$

(minimiser $\|w\|_1$, $w \in \mathbb{R}^d$ ↳ argmin : {0} ↳ min = 0)

$$\text{minimiser}_{w \in \mathbb{R}^d} \; f(w) + \lambda \|w\|_1 \quad \text{pour} \quad \lambda \geq 0$$

But: Pénaliser les vecteurs qui n'ont pas beaucoup de coefficients nuls / Favoriser les vecteurs creux / "sparse"

Remarque: la régularisation "idéale" pour avoir une solution creuse serait $\lambda \|w\|_0$, $\|w\|_0 =$ { nombre de coefficients non nuls de w }



"Norme ℓ_0"

↳ Non convexe, discontinue, à valeurs discrètes (constante par morceaux)

⟶ Norme ℓ_1 est convexe, continue et c'est la fonction convexe la plus proche de $\|\|_0$ autour de 0

Illustration :

$$\underset{w \in \mathbb{R}^2}{\text{minimiser}} \; \|Xw - y\|^2 + \lambda \|w\|_1$$

$$X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad y = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$\lambda = 0$ $\quad \underset{w \in \mathbb{R}^2}{\text{minimiser}} \; (w_1 - 1)^2 + (w_2 - 2)^2$

$$\text{argmin}(\;) = \left\{ \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right\}$$

$\lambda > 0$ $\quad \underset{}{\text{minimiser}} \; (w_1 - 1)^2 + (w_2 - 2)^2 + \underbrace{\lambda \|w\|_1}_{\lambda |w_1| + \lambda |w_2|}$

$0 \in 2(w_1 - 1) + \partial \|w\|_1$
$-2(w_1 - 1)$

↳ Via les sous-gradients, on montre que le problème a une unique solution

$$0 \leq \lambda < 1 \qquad \begin{bmatrix} 1 - \lambda \\ 2 - \lambda \end{bmatrix}$$

$$1 \leq \lambda < 2 \qquad \begin{bmatrix} 0 \\ 2 - \lambda \end{bmatrix}$$

$$2 \leq \lambda \qquad \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

→ la régularisation l_1 permet d'identifier les coordonnées les plus importantes en faisant varier λ. Plus une coordonnée sera importante, plus λ devra être grand pour qu'elle soit nulle en la solution

⇒ A l'origine, la régularisation l_1 était utilisée en régression linéaire pour de la sélection d'attributs (feature selection)

En vertus d'algorithmes

- Si f est convexe, alors $f + \lambda\|\cdot\|_2$ aussi et on peut utiliser une méthode de sous-gradient
- Pour f C^1, il existe une méthode dédiée (ISTA) qui est une variante de l'algorithme du gradient proximal

↳ Il existe de nombreux autres termes de régularisation basés sur la norme ℓ_1 qui encouragent la parcimonie de manière structurée

Ex) Group LASSO

$$r(w) = \sum_{g \in G} \|w_g\| = \left\| \begin{bmatrix} \|w_{g_1}\| \\ \vdots \\ \|w_{g_m}\| \end{bmatrix} \right\|_1 \quad \text{où} \quad w = \begin{bmatrix} w_{g_1} \\ \vdots \\ w_{g_m} \end{bmatrix}$$

M groupes de variables

$G = \{g_1, \dots, g_m\}$

Toutes les variables d'un même groupe doivent être soit nulles, soit non nulles

(3) Méthodes proximales / Gradient proximal

Idée : On cherche un algorithme qui puisse s'appliquer aux problèmes de la forme suivante

$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \; f(w) + \lambda r(w) \qquad \lambda \geq 0$$

Hypothèses :

- f C^1 (mais pas forcément convexe)
- r convexe (mais pas forcément C^1)

Algorithme du gradient proximal pour ce problème

$$w_{k+1} \in \underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ f(w_k) + \nabla f(w_k)^T (w - w_k) + \frac{1}{2\alpha_k} \|w - w_k\|^2 + \lambda r(w) \right\}$$

avec $\alpha_k > 0$

- $f(w_k) + \nabla f(w_k)^T (w - w_k)$: Approximation de $f(w)$ autour de w_k
- $\frac{1}{2\alpha_k} \|w - w_k\|^2$: Terme proximal pénalise les w loin de w_k
- $\lambda r(w)$: Terme de régularisation (inchangé par rapport au problème de base)

Une itération de gradient proximal :

- calcule une solution d'un problème d'optimisation "sous-problème" où on a "linéarisé" le terme $f(w)$
- pour résoudre le sous-problème (une fois que $\nabla f(w_k)$ est calculé) il n'est pas nécessaire d'accéder à f ou à son gradient (et donc aux données)
- le sous-problème est toujours fortement convexe, et donc w_{k+1} est défini de manière unique

L'algorithme du gradient proximal n'est intéressant que si le sous-problème est facile à résoudre (plus facile au moins que le problème d'origine). Cela est souvent le cas lorsque le "prox" de r est facile à calculer

Def : $h : \mathbb{R}^d \to \mathbb{R}$ convexe

$$\forall w \in \mathbb{R}^d, \quad \operatorname{prox}_h(w) \in \underset{v \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ \frac{1}{2} \|v - w\|^2 + h(v) \right\}$$

($\frac{1}{2}\|v - w\|^2 + h(v)$: fortement convexe)

L'itération du gradient proximal se réécrit

$$w_{k+1} \in \operatorname{argmin} \left\{ f(w_k) + \nabla f(w_k)^T (w - w_k) + \frac{1}{2\alpha_k} \|w - w_k\|^2 + \lambda r(w) \right\}$$

$$\Leftrightarrow \quad w_{k+1} = \text{prox}_{\alpha_k r}\left(w_k - \alpha_k \nabla f(w_k) \right)$$

$w_k - \alpha_k \nabla f(w_k)$: itération de la descente de gradient

$$\text{prox}_{\alpha r}(w) \in \underset{v}{\text{argmin}} \left\{ \frac{1}{2}\|v-w\|^2 + \alpha r(v) \right\}$$

Exemples

- $r(w) = 0$ (ou $\lambda = 0$)

$$\text{prox}_0(w) \in \underset{v}{\text{argmin}} \left\{ \frac{1}{2}\|v-w\|^2 + 0 \right\} = \{w\}$$

Sans régularisation, l'itération de gradient proximal devient $w_{k+1} = w_k - \alpha_k \nabla f(w_k)$ $\Rightarrow$ Descente de gradient !

- $r(w) = \frac{1}{2}\|w\|^2$

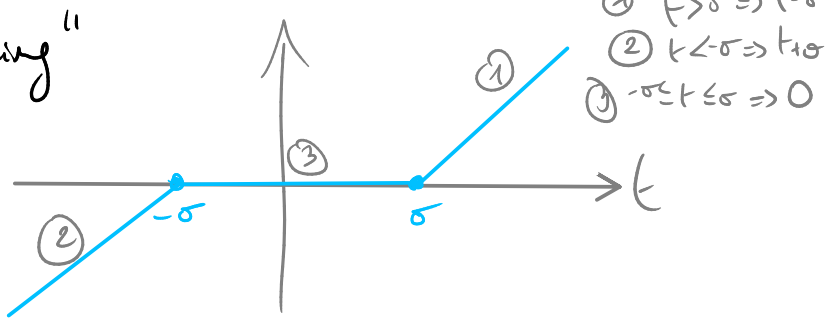
$$w_{k+1} = \text{prox}_{\alpha_k r}(w_k - \alpha_k \nabla f(w_k)) = \frac{1}{1+\lambda\alpha_k} w_k - \frac{\alpha_k}{1+\lambda\alpha_k} \nabla f(w_k)$$

$\frac{1}{1+\lambda\alpha_k}$: "weight decay" ; $\frac{\alpha_k}{1+\lambda\alpha_k}$: "gradient decay"

$\neq$ descente de gradient appliquée au problème avec régularisation $\left(w_{k+1} = (1-\lambda\alpha_k) w_k - \alpha_k \nabla f(w_k) \right)$

- $r(w) = \|w\|_1$

$\hookrightarrow$ Le "prox" de la norme ℓ_1 est défini par une fonction dite de "soft-thresholding"



Pour le gradient proximal, on obtient

$$\forall j = 1..d, \quad [w_{k+1}]_j = \begin{cases} [w_k - \alpha_k \nabla f(w_k)]_j - \lambda \alpha_k & \text{si } [w_k - \alpha_k \nabla f(w_k)]_j > \lambda \alpha_k \\ [w_k - \alpha_k \nabla f(w_k)]_j + \lambda \alpha_k & \text{si } [w_k - \alpha_k \nabla f(w_k)]_j < -\lambda \alpha_k \\ 0 & \text{sinon} \end{cases}$$

→ Cette formule produit une suite d'itérés qui ont nécessairement au moins autant de coordonnées nulles que les itérés de la descente de gradient

$$\forall k, \quad \|w_{k+1}\|_0 \leq \|w_k - \alpha_k \nabla f(w_k)\|_0$$

→ la méthode du gradient proximal avec régularisation ℓ_1 est connue (notamment en traitement du signal) comme la méthode ISTA (Iterative Soft-Thresholding Algorithm)

NB: ISTA avec momentum = FISTA

(Le gradient proximal peut être décliné en versions stochastique, avec sous-gradients, etc)

→ Cas d'application principal de ISTA / régularisation ℓ_1:

"Sparse recovery"

$(X, y) \in \mathbb{R}^{n \times d} \times \mathbb{R}^n$

$d >> n$

$y = X \overbrace{w^*}^{\text{"ground truth"}} + \varepsilon$

$\|w^*\|_0 << d$

ε : bruit (par ex. gaussien)

But: Parmi tous les modèles possibles (w), trouver celui le plus parcimonieux qui colle aux données

$\Rightarrow$ minimiser $\frac{1}{2}\|Xw-y\|^2 + \lambda\|w\|_1$, résolu par ISTA/gradient proximal

Autres exemples en traitement du signal et des images

- **Débruitage**

z : image bruitée $(x^* + \varepsilon)$

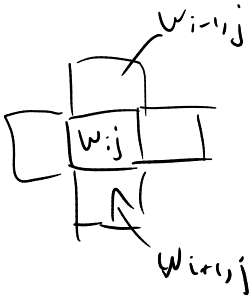
Postulat : Pour une image bruitée, la différence entre des pixels voisins doit être plus grande que pour une image non bruitée

$\rightarrow$ Pour débruiter l'image, on peut alors calculer le "prox" suivant

$$\text{prox}_{\lambda TV}(z) = \underset{w}{\text{argmin}} \left\{ \frac{1}{2}\|w-z\|^2 + \lambda\|w\|_{TV} \right\}$$

$\|w\|_{TV}$: Variation totale

$$\|w\|_{TV} = \sum_{(i,j)} \|(\nabla w)_{ij}\| \qquad (i,j) \text{ indice pixel}$$



$$(\nabla w)_{ij} = (w_{ij} - w_{i,j+1})^2 + (w_{ij} - w_{i,j-1})^2 + (w_{ij} - w_{i+1,j})^2 + (w_{ij} - w_{i-1,j})^2$$

High-dimensional data analysis with low-dimensional models
J. Wright, Y. Ma (2022)

Approximation de matrice

$X \in \mathbb{R}^{n \times d}$: image (donnée)

But : Récupérer une structure sous-jacente à l'image

1. $X = L + E$, L matrice de rang faible $r \ll \min(n,d)$
et E une perturbation de rang plein
$\text{rang}(E) = \min(n,d)$

Pb : Retrouver L à partir de X

Formulation : $\underset{W \in \mathbb{R}^{n \times d}}{\text{minimiser}} \quad \frac{1}{2}\|W - X\|_F^2 + \lambda \|W\|_*$

$\|W-X\|_F^2 = \sum_{ij} (W_{ij} - X_{ij})^2$ "Norme de Frobenius"

$\|W\|_*$ Norme nucléaire (pénalise les matrices de rang élevé)

$\|W\|_* =$ somme des valeurs singulières $\sigma_1 \geq \dots \geq \sigma_{\text{rang}(W)}$

$\forall A \in \mathbb{R}^{n \times d}$,

A rang $r \iff A = U \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_r \end{bmatrix} V^T$ (SVD de A)

avec $\sigma_1 \geq \dots \geq \sigma_r > 0$

$U \in \mathbb{R}^{n \times r}$, $U^T U = I_r$

$V \in \mathbb{R}^{d \times r}$, $V^T V = I_r$

→ On peut calculer explicitement le prox de $\|.\|_*$

→ Donc on peut faire du gradient proximal !

$I_n = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{bmatrix}$

$\|I_n\|_1 = \sum_{ij} |(I_n)_{ij}| = n < n^2$

$\frac{n}{n^2}$ coeffs non nuls

$\|I_n\|_* = n$) rang maximal

2) Approximation de rang faible + parcimonieuse

Hypothèse : $X = \underbrace{L}_{\text{rang}(L) \ll \min(m,d)} + S$ ← parcimonieuse, $\|S\|_0 \ll md$

$$\underset{\substack{W \in \mathbb{R}^{m \times d} \\ U \in \mathbb{R}^{m \times d}}}{\text{minimiser}} \quad \frac{1}{2} \|X - (W+U)\|_F^2 + \lambda \underbrace{\|W\|_*}_{\text{rang faible}} + \mu \underbrace{\|U\|_1}_{\text{parcimonie}}$$