OUTILS D'OPTIMISATION POUR LES SCIENCES DES DONNÉES ET DE LA DÉCISION

10 novembre 2023

Dernière séance :

→ Fin cours optimisation et décomposition

→ TD : Examen 2022-2023 (partie 2)

(3) Décomposition)

Problème :

$$(P)\begin{cases} \underset{\substack{u \in \mathbb{R}^{d_1} \\ v \in \mathbb{R}^{d_2}}}{\text{minimiser}} \ f(u) + g(v) & \text{s.c.} \quad Au + Bv = c \end{cases}$$

$f : \mathbb{R}^{d_1} \to \mathbb{R}$

$g : \mathbb{R}^{d_2} \to \mathbb{R}$

$A \in \mathbb{R}^{m \times d_1}$, $B \in \mathbb{R}^{m \times d_2}$

$c \in \mathbb{R}^m$

Problème séparable :

- Objectif : somme de deux fonctions de groupes de variables distincts
- Contraintes : définies via deux fonctions de ces mêmes groupes de variables

Lagrangien du problème

Rappel : Pour $\underset{w \in \mathbb{R}^d}{\text{minimiser}} \ f(w)$ s.c. $Aw = b$, le lagrangien s'écrit

$$\mathcal{L}(w, z) = f(w) + z^T (Aw - b)$$

Le lagrangien associé à (P) est la fonction

$$\mathcal{L} : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \times \mathbb{R}^m \to \mathbb{R}$$

$$(\underbrace{u, v}_{\substack{\text{Variables} \\ \text{du problème} \\ (P)}}, \underbrace{z}_{\substack{\text{variable duale/} \\ \text{multiplicateur de Lagrange}}}) \longmapsto \mathcal{L}(u, v, z) = f(u) + g(v) + z^T (Au + Bv - c)$$

. ↳ Comme on a défini un lagrangien, on peut également définir des méthodes d'optimisation duales pour résoudre (P)

* Montée duale ⟹ Décomposition duale
* Lagrangien augmenté ⟹ ADMM

Décomposition duale

- Initialisation : $u_0 \in \mathbb{R}^{d_1}$, $v_0 \in \mathbb{R}^{d_2}$, $z_0 \in \mathbb{R}^m$
- Pour $k = 0, 1, \ldots$
  - Calculer $u_{k+1} \in \operatorname{argmin}_{u \in \mathbb{R}^{d_1}} \mathcal{L}(u, v_k, z_k)$] Optimisation selon u
  - Calculer $v_{k+1} \in \operatorname{argmin}_{v \in \mathbb{R}^{d_2}} \mathcal{L}(u_k, v, z_k)$] Optimisation selon v (u, z fixés)
  - Calculer $z_{k+1} = z_k + \alpha_k (A u_{k+1} + B v_{k+1} - c)$ $\qquad \alpha_k > 0$

Mise à jour de la variable duale (pas de sous-gradient pour le problème dual)

Intérêt pratique

- Les calculs de u_{k+1} et v_{k+1} sont indépendants et donc peuvent être effectués en parallèle
- Par rapport à l'approche duale générique, on résout deux problèmes de taille d_1 et d_2 au lieu d'un problème de taille $d_1 + d_2$, ce qui est souvent plus efficace en termes de calcul

ADMM (Alternating Direction Method of Multipliers)

Méthode des multiplicateurs : autre nom de l'algorithme du lagrangien augmenté

Lagrangien augmenté associé à (P)

$$\mathcal{L}^a : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \times \mathbb{R}^m \times \mathbb{R}_{++} \longrightarrow \mathbb{R}$$

$$(u, v, z, \lambda) \mapsto \mathcal{L}^a(u, v, z; \lambda)$$

avec

$$\mathcal{L}^a(u,v,z\,;\lambda) = \underbrace{f(u)+g(v)+z^T(Au+Bv-c)}_{\mathcal{L}(u,v,z)} + \frac{\lambda}{2}\|Au+Bv-c\|^2$$

Régularisation
$\Rightarrow$ pénalisation des couples (u,v) qui ne vérifient pas la contrainte

Algorithme

- Initialisation : $u_0 \in \mathbb{R}^{d_1}$, $v_0 \in \mathbb{R}^{d_2}$, $z_0 \in \mathbb{R}^m$, $\lambda > 0$.
- Pour $k = 0, 1, \dots$
  - Calculer $u_{k+1} \in \underset{u \in \mathbb{R}^{d_1}}{\operatorname{argmin}} \mathcal{L}^a(u, v_k, z_k\,;\lambda)$
  - Calculer $v_{k+1} \in \underset{v \in \mathbb{R}^{d_2}}{\operatorname{argmin}} \mathcal{L}^a(u_k, v, z_k\,;\lambda)$
  - Calculer $z_{k+1} = z_k + \lambda(Au_{k+1} + Bv_{k+1} - c)$

Commentaires

$\rightarrow$ On peut choisir d'autres tailles de pas que λ dans le calcul de z_{k+1}

$\rightarrow$ On peut faire varier λ au cours des itérations $\{\lambda_k\}$ (en général $\lambda_{k+1} \geq \lambda_k$)

Remarque : Toutes les idées ci-dessus s'adaptent à une décomposition en un nombre quelconque de vecteurs de variables

$$\underset{\substack{u_1 \in \mathbb{R}^{d_1} \\ \vdots \\ u_q \in \mathbb{R}^{d_q}}}{\text{minimiser}} \sum_{i=1}^{q} f_i(u_i) \quad \text{s.c.} \quad \sum_{i=1}^{q} A_i u_i = b$$

$f_i : \mathbb{R}^{d_i} \to \mathbb{R} \quad \forall i$

$A_i \in \mathbb{R}^{m \times d_i} \quad \forall i$

$b \in \mathbb{R}^m$

Exemple : "Group LASSO"

LASSO : $X \in \mathbb{R}^{m \times d}$, $y \in \mathbb{R}^m$

$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \ \frac{1}{2}\|Xw - y\|^2 + \lambda \|w\|_1$$

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

$\Rightarrow$ Solution est parcimonieuse (beaucoup de coefficients nuls)

$\|v\| = \sqrt{\sum v_j^2}$

Group LASSO

$$\underset{w \in \mathbb{R}^d}{\text{minimiser}} \ \frac{1}{2}\|Xw - y\|^2 + \lambda \sum_{\ell=1}^{L} \|w_{[\ell]}\|$$

avec $w = \begin{bmatrix} w_{[1]} \\ \vdots \\ w_{[L]} \end{bmatrix}$

$L = d$ LASSO

$L < d \Rightarrow$ donne des solutions avec beaucoup de sous-vecteurs $w_{[\ell]}$ nuls

① On introduit un vecteur de variables $z \in \mathbb{R}^d$ pour rendre le problème séparable

② On force w et z à être égaux via une contrainte linéaire

$$w - z = 0 \iff \sum_{\ell=1}^{L} (w_{[\ell]} - z_{[\ell]}) = 0$$

$\Rightarrow$ On obtient

$$\underset{\substack{w \in \mathbb{R}^d \\ z \in \mathbb{R}^d}}{\text{minimiser}} \ \frac{1}{2}\|Xw - y\|^2 + \lambda \sum_{\ell=1}^{L} \|z_{[\ell]}\| \quad \text{s.c.} \quad \sum_{\ell=1}^{L} (w_{[\ell]} - z_{[\ell]}) = 0$$

(−) On a ajouté des variables

(+) On a obtenu un problème séparable en w et z et sur ce problème les itérations d'ADMM en w et z sont plus simples à calculer qu'une itération de gradient proximal sur le problème de départ

↳ On peut même introduire encore plus de séparabilité

$$\underset{\substack{w \in \mathbb{R}^d \\ \underbrace{v_1}_{d_1}, \dots, \underbrace{v_L}_{d_L}}}{\text{minimiser}} \ \frac{1}{2}\|Xw - y\|^2 + \lambda \sum_{l=1}^{L} \|v_l\| \quad \text{s.c.} \quad w[l] - v_l = 0 \quad l = 1..L$$

$d_1 + \dots + d_L = d$

NB: Très grand nombre de variantes sur ADMM :
- stochastique
- proximal
- accélération (sur certains problèmes convexes)

④ Optimisation par consensus et décentralisée

↳ Contexte : m entités/agents qui possèdent chacun leur fonction objectif $f_i : \mathbb{R}^d \to \mathbb{R}$ qui n'est pas visible par les autres agents

But : $\underset{w \in \mathbb{R}^d}{\text{minimiser}} \ \sum_{i=1}^{m} f_i(w)$

↳ Modélisation : On considère que chaque agent dispose d'une copie des variables $w^{(i)} \in \mathbb{R}^d$ et on optimise donc la somme des $f_i(w^{(i)})$

$\Rightarrow$ On impose (via des contraintes) le fait que les agents doivent arriver à un consensus, càd que $w^{(i)} = w^{(j)} \quad \forall (i,j)$

Deux problèmes d'optimisation possibles :

① $$\underset{\substack{w \in \mathbb{R}^d \\ w^{(1)} \in \mathbb{R}^d \\ \vdots \\ w^{(m)} \in \mathbb{R}^d}}{\text{minimiser}} \underbrace{\sum_{i=1}^{m} f_i(w^{(i)})}_{\text{Fonction à optimiser}} \quad \text{s.c.} \quad \underbrace{w - w^{(i)} = 0}_{\text{Contrainte de consensus}} \quad \forall i \in \{1..m\}$$

Optimisation par consensus en distribué avec une variable "maîtresse" w partagée par tous les agents

$\Rightarrow$ On peut appliquer une technique de décomposition avec $u = \begin{bmatrix} w^{(1)} \\ \vdots \\ w^{(m)} \end{bmatrix}$ et $v = w$

(on peut décomposer en $u^{(1)} = w^{(1)}, \dots, u^{(m)} = w^{(m)}, u^{(m+1)} = w$)

$\Rightarrow$ Le problème à résoudre pour mettre à jour w est très simple

② $$\underset{\substack{w^{(1)} \in \mathbb{R}^d \\ \vdots \\ w^{(m)} \in \mathbb{R}^d}}{\text{minimiser}} \sum_{i=1}^{m} f_i(w^{(i)}) \quad \text{s.c.} \quad w^{(i)} - w^{(j)} = 0 \quad \forall 1 \leq i < j \leq m$$

Problème d'optimisation décentralisé (pas de variable primale partagée entre les agents) $\Rightarrow$ les techniques duales s'appliquent encore

↳ Ce concept d'optimisation décentralisée se généralise encore plus : à un réseau de m agents qui communiquent via un graphe

$$V = \{1, \dots, m\} \quad \text{sommets}$$

$$\mathcal{E} \subseteq \{1, \dots, m\}^2 \quad \text{arêtes}$$

(3) $$\underset{\substack{w^{(1)} \in \mathbb{R}^d \\ \vdots \\ w^{(m)} \in \mathbb{R}^d}}{\text{minimiser}} \sum_{i=1}^{m} f_i(w^{(i)}) \quad \text{s.c.} \quad w^{(i)} - w^{(j)} = 0 \quad \forall (i,j) \in \mathcal{E}$$

"Consensus entre voisins"