Optimisation distribuée pour la résolution d'un problème de *clustering*

Igor Colin

23 mai 2014

1 Description du problème

1.1 Problème initial

Soit $\mathcal{X}$ un espace de dimension finie et soit $(X_i)_{1\leq i\leq n}$ n points dans $\mathcal{X}$. Pour $K > 0$, on cherche à résoudre le problème de clustering suivant [Clémençon, 2014] :

$$\min_{P\in\mathcal{P}_K} f(P) = \frac{1}{n^2}\sum_{1\leq i,j\leq n} D(X_i, X_j)\Phi_P(i,j) \tag{1}$$

où $\mathcal{P}_K$ est l'ensemble des partitions de taille K de $\{1,\dots,n\}$, D est une mesure de dissimilarité entre les éléments de $\mathcal{X}$ et Φ_P est définie pour tout $P \in \mathcal{P}_K$ par :

$$\Phi_P : \begin{cases} \{1,\dots,n\}^2 & \to & \{0,1\} \\ (i,j) & \mapsto & \begin{cases} 1 & \text{si } i \text{ et } j \text{ sont dans le même } \textit{cluster} \\ 0 & \text{sinon} \end{cases} \end{cases}$$

La mesure de dissimilarité D doit vérifier les propriétés suivantes :
Symétrie : Pour $(x, x') \in \mathcal{X}^2$, $D(x, x') = D(x', x)$
Séparation : Pour $(x, x') \in \mathcal{X}^2$, $D(x, x') = 0$ si et seulement si $x = x'$

On remarque qu'en définissant, pour $1 \leq i \leq n$,

$$f_i : \begin{cases} \mathcal{P}_K & \to & \mathbb{R}_+ \\ P & \mapsto & \frac{1}{n}\sum_{j=1}^n D(X_i, X_j)\Phi_P(i,j) \end{cases}$$

on peut réécrire le problème (1) sous la forme :

$$\min_{P\in\mathcal{P}_K} \frac{1}{n}\sum_{i=1}^n f_i(P) \tag{2}$$

Dans cette formulation, nous avons seulement décomposé la fonction objectif f en n fonctions objectifs partielles.

1.2 Contraintes supplémentaires

On souhaite résoudre le problème de *clustering* précédent de manière distribuée. On considère pour cela un graphe non-orienté $G = (V, E)$, où $V = \{1,\dots,n\}$. Chaque sommet $i \in V$ du graphe représente un agent, capable de calculer uniquement les valeurs de f_i sur $\mathcal{P}_K$. On considère également que les différents agents sont capables de communiquer (i.e. donner la valeur de leur fonction objectif partielle en un point), avec leur voisinage. C'est-à-dire que deux agents $(i, j) \in V^2$ pourront communiquer entre eux si et seulement si $(i, j) \in E$.

2 Résolution du problème

2.1 Algorithme de moyennage de dual distribué

Les méthodes d'optimisation ou de classification distribuée sont nombreuses et variées. Afin de pouvoir facilement adapter la méthode à notre problème, on se penchera sur des techniques génériques, demandant peu d'hypothèses sur la structure du problème. On souhaite également utiliser des algorithmes possédant de bonnes garanties théoriques concernant leur vitesse de convergence ainsi que leur précision. Pour ces raisons, nous nous concentrerons par la suite sur l'algorithme de moyennage de dual distribué [Duchi et al., 2012, Nesterov, 2009, Xiao et al., 2010].

Cette méthode est adaptée de la méthode de moyennage de dual visant à résoudre le problème d'optimisation suivant :

$$\min_{x \in \mathcal{X}} f(x) \tag{3}$$

où $\mathcal{X} \subset \mathbb{R}^d$ est un fermé convexe et f est une fonction convexe, sous-différentiable sur $\mathcal{X}$ et L-lipschitzienne pour un certain $L > 0$. Le moyenne de dual repose fortement sur l'utilisation d'un opérateur proximal $\psi : \mathcal{X} \to \mathbb{R}$, fortement convexe sur $\mathcal{X}$ par rapport à une certaine norme N. Le choix du couple (ψ, N) s'effectuera en fonction du problème considéré. Par exemple, si $\mathcal{X}$ est le simplexe de $\mathbb{R}^d$, on aura tendance à choisir la fonction d'entropie pour ψ et la norme-1 pour N. L'idée de la méthode, présentée dans l'Algorithme 1, est de générer une suite $(z_t, x_t)_{t>0}$ telle que :

1. $z_{t+1} = z_t + g$, où g est un élement de $\partial f(x_t)$.
2. $x_{t+1} = \arg\min_{x \in \mathcal{X}} \left\{ \langle z_{t+1}, x \rangle + \frac{1}{\alpha_t} \psi(x) \right\}$, où $(\alpha_t)_{t \geq 0}$ est une suite décroissante d'éléments positifs.

Algorithm 1 Algorithme du moyennage de dual.

Require: $\psi, N, \alpha, x_0, z_0$
  $t \leftarrow 0$
  while not convergence **do**
    $g \in \partial f(x_t)$
    $z_{t+1} \leftarrow z_t + g$
    $x_{t+1} \leftarrow \arg\min_{x \in \mathcal{X}} \left\{ \langle z_{t+1}, x \rangle + \frac{1}{\alpha_t} \psi(x) \right\}$
    $t \leftarrow t + 1$
  end while

Dans l'adaptation distribuée de cette méthode, on reformule le problème (3) ainsi :

$$\min_{x \in \mathcal{X}} \frac{1}{n} \sum_{i=1}^{n} f_i(x) \tag{4}$$

où les $(f_i)_{1 \leq i \leq n}$ sont des fonctions convexes, sous-différentiables et L-lipschitzienne sur $\mathcal{X}$ pour un certain $L > 0$. L'objectif est désormais de générer des suites $(z_t^{(i)}, x_t^{(i)})_{t>0}$, pour $1 \leq i \leq n$. Pour cela, les étapes de mise à jour sont légèrement modifiées (c.f. Algorithme 2) :

1. $z_{t+1}^{(i)} = \sum_{j=1}^{n} c_{ij} z_t^{(j)} + g$, où $C = (c_{kl})_{1 \leq k,l \leq n}$ est une matrice doublement stochastique, et $g \in \partial f(x_t^{(i)})$.
2. $x_{t+1}^{(i)} = \arg\min_{x \in \mathcal{X}} \left\{ \langle z_{t+1}^{(i)}, x \rangle + \frac{1}{\alpha_t} \psi(x) \right\}$

Algorithm 2 Algorithme du moyennage de dual distribué.

Require: $\psi, N, C, \alpha, x_0, z_0$
  $t \leftarrow 0$
  while not convergence **do**
    for $i \in 1, \dots, n$ **do**
      $g \in \partial f(x_t)$
      $z_{t+1}^{(i)} \leftarrow \sum_{j=1}^{n} z_t^{(j)} + g$
      $x_{t+1}^{(i)} \leftarrow \arg\min_{x \in \mathcal{X}} \left\{ \langle z_{t+1}^{(i)}, x \rangle + \frac{1}{\alpha_t} \psi(x) \right\}$
    end for
    $t \leftarrow t + 1$
  end while

2.2 Reformulation du problème

La méthode du moyennage de dual distribuée est plutôt générique puisqu'elle ne nécessite qu'une hypothèse de convexité et de différentiabilité sur les fonctions manipulées. Malheureusement, le problème tel qu'il est formulé en (2) n'est pas convexe. Une approche naturelle pour contourner cette difficulté est de rendre l'appartenance aux *clusters* « continue ». Dans la formulation initiale, cette appartenance est binaire : un élément de l'échantillon appartient ou n'appartient pas à un certain *cluster*. On considère donc le problème suivant :

$$\min_{(a_l)_{1 \leq l \leq n} \in (\tilde{\Delta}^K)^n} \quad \frac{1}{n^2} \sum_{1 \leq i,j \leq n} D(X_i, X_j) \langle a_i, a_j \rangle \tag{5}$$

où $\tilde{\Delta}^K = \left\{ x \in \Delta^K, \|x\|_1 = 1 \right\}$. Désormais, à chaque élément de l'échantillon est associé un vecteur de poids, représentant la probabilité pour cet élément d'appartenir à un *cluster* donné. On remarque que si l'on remplace $\tilde{\Delta^K}$ par $\tilde{\Delta^K} \cap \{0,1\}^K$, on se ramène bien au problème (2).

Le problème d'optimisation (5) est convexe, sous réserve que la matrice $D = \left(D(X_i, X_j)\right)_{1 \leq i,j \leq n}$ soit semi-définie positive.

Références

[Clémençon, 2014] Clémençon, S. (2014). A statistical view of clustering performance through the theory of u-processes. *Journal of Multivariate Analysis*, 124 :42–56.

[Duchi et al., 2012] Duchi, J. C., Agarwal, A., and Wainwright, M. J. (2012). Dual averaging for distributed optimization : convergence analysis and network scaling. *Automatic Control, IEEE Transactions on*, 57(3) :592–606.

[Nesterov, 2009] Nesterov, Y. (2009). Primal-dual subgradient methods for convex problems. *Mathematical programming*, 120(1) :221–259.

[Xiao et al., 2010] Xiao, L. et al. (2010). Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11(2543-2596) :4.

3 Résultats

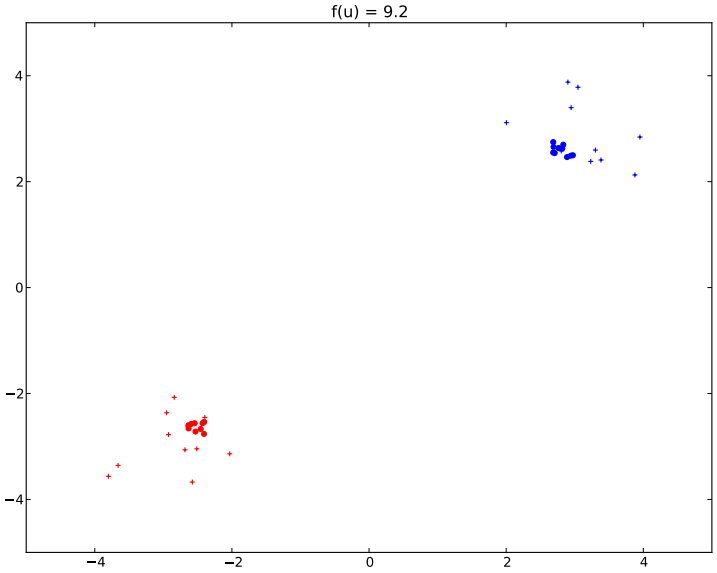

FIGURE 1 – Clustering utilisant la version non distribuée de l'algorithme.

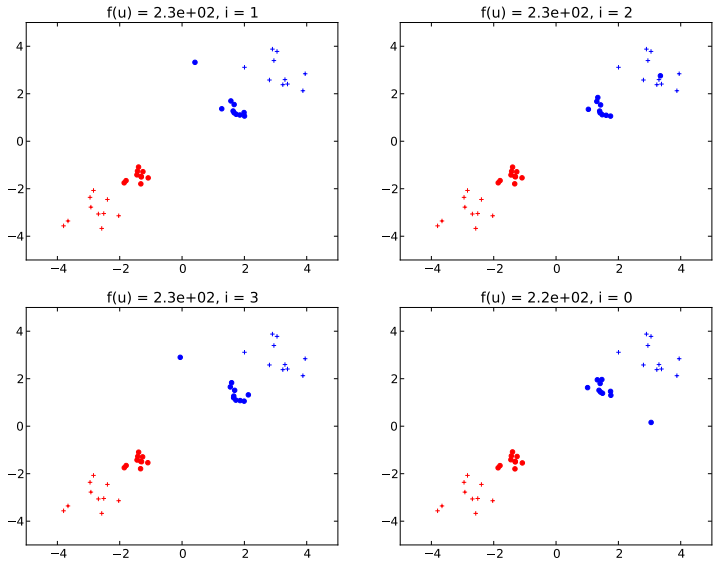

FIGURE 2 – Clustering utilisant la version distribuée de l'algorithme

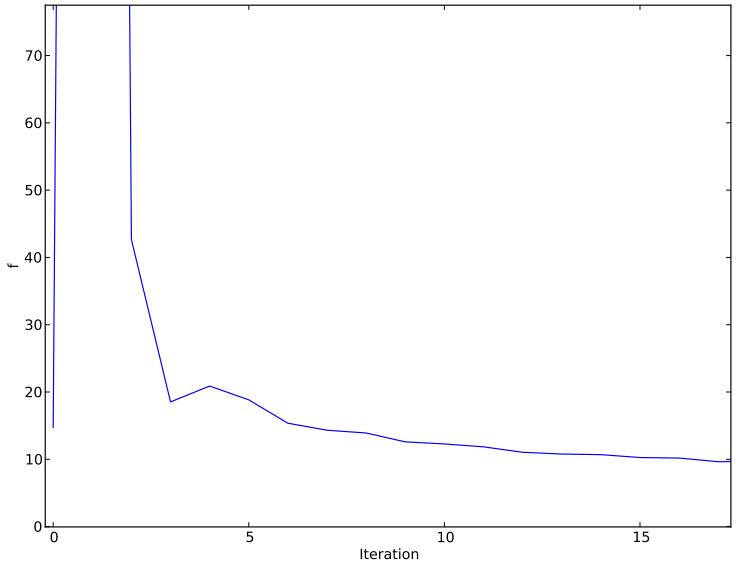

FIGURE 3 – Valeurs de la fonction objectif avec les itérations (version non-distribuée).

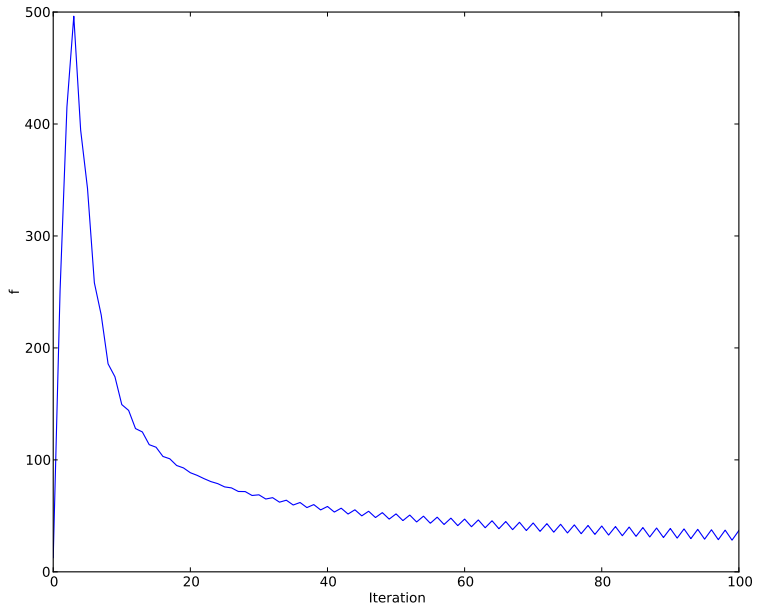

FIGURE 4 – Valeurs d'une fonction objectif partielle avec les itérations.

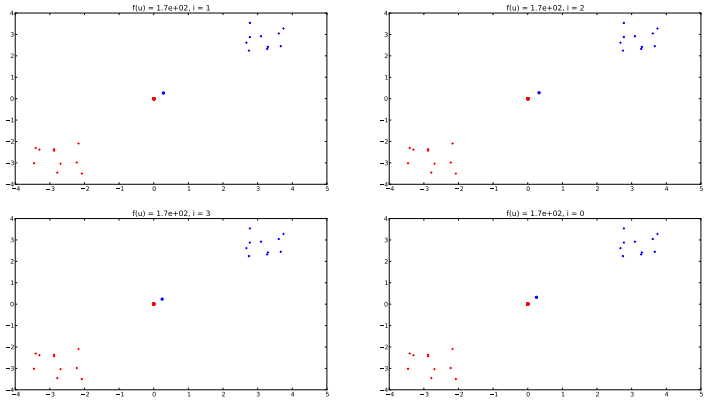

FIGURE 5 – Clustering distribué pour un graphe partiellement connecté (50/50)

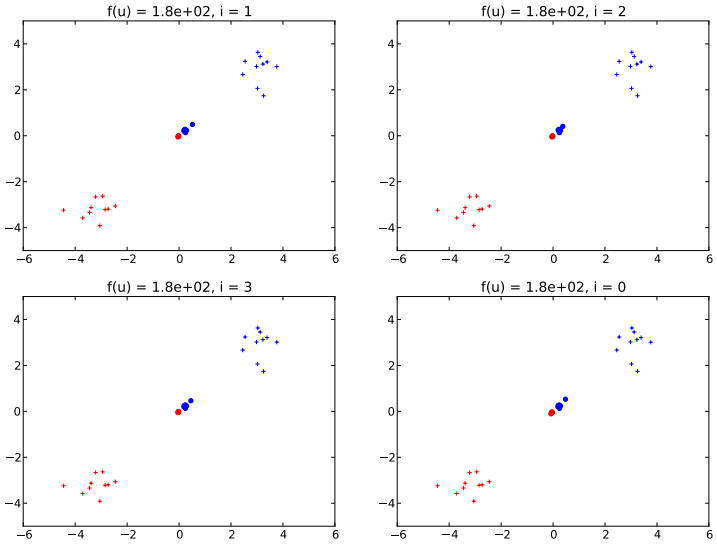

FIGURE 6 – Clustering distribué pour un graphe partiellement connecté (100/0)