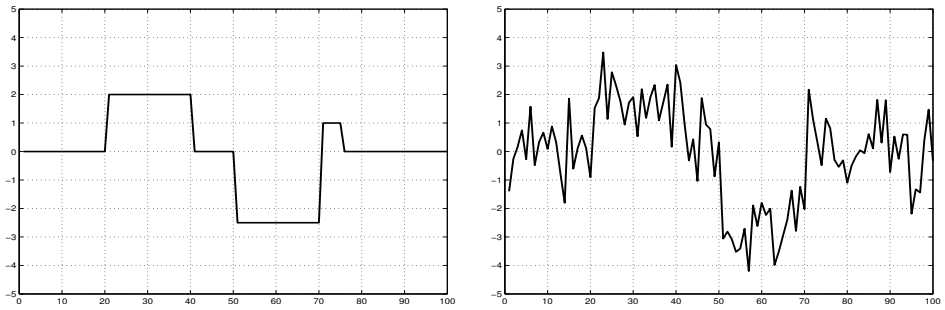
TP : Optimisation convexe non-lisse

1 Problème de débruitage

Soit $\overline{x} = (\overline{x}^{(i)})_{1 \leq i \leq N} \in \mathbb{R}^N$ un signal échantillonné constant par morceaux. On note $y = \overline{x} + \varepsilon$ une version bruitée de $\overline{x}$ avec $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Un exemple de signaux $\overline{x}$ et y est présenté Figure 1.



Signal original $\overline{x}$ Signal bruitée y

FIGURE 1 – Exemple de signal constant par morceaux de taille $N = 100$ bruité par un bruit blanc Gaussien de variance $\sigma^2 = 1$.

Le but de ce premier exercice est d'estimer un signal $\widehat{x}$ constant par morceaux au plus proche de $\overline{x}$ à partir des observations y. Une solution consiste à minimiser le critère suivant :

$$\widehat{x}_\lambda = \arg\min_{x \in \mathbb{R}^N} \frac{1}{2}\|x - y\|_2^2 + \lambda \|Lx\|_1$$

où $(Lx)^{(i)} = x^{(i+1)} - x^{(i)}$ pour tout $i \in \{1, \ldots, N-1\}$ et $\lambda > 0$ est le paramètre de régularisation. $L \in \mathbb{R}^{N \times N}$ désigne un opérateur de différences finies.

1) Discuter de l'influence du paramètre λ sur la solution $\widehat{x}_\lambda$.

2) Montrer que le problème dual associé peut s'écrire

$$\widehat{u}_\lambda \in \mathrm{Argmin}_{u \in \mathbb{R}^N} \frac{1}{2}\|y + L^* u\|_2^2 \quad \text{s.t.} \quad \|u\|_\infty \leq \lambda,$$

et que la relation entre les solutions primale et duale est

$$\widehat{x}_\lambda = y + L^* \widehat{u}_\lambda.$$

3) Résolution du problème dual à l'aide de l'algorithme explicite-implicte (forward-backward) présenté en cours.

- Quelle est la fonction différentiable de gradient Lipschitz ? Préciser son gradient et sa constante de Lipschitz.

- En utilisant la relation qui relie l'opérateur proximal d'une fonction $f \in \Gamma_0(\mathbb{R}^N)$ et sa fonction conjuguée f^*, i.e.,

$$(\forall x \in \mathbb{R}^N) \quad \mathrm{prox}_{\gamma f^*} x = x - \gamma \mathrm{prox}_{\gamma^{-1} f}(\gamma^{-1} x),$$

  donner l'expression de $P_{\|\cdot\|_\infty \leq \lambda}$.

- Dans MATLAB, charger le signal signal_exo1.mat

  >> load signal_ex1.mat ;

- Créer une version bruitée de ce signal

  >> y = x + randn(size(x)) ;

- Implémenter l'algorithme explicite-implicte et en déduire $\widehat{x}$. Utiliser les fonctions opL.m, opL_adj.m, opL.m permettant de calculer respectivement T, T^* et l'opérateur proximal associé à la norme ℓ_1.

- Tracer l'évolution de la fonction objectif primale et celle de la fonction objectif duale en fonction des itérations. Faire varier les paramètres de l'alorithmes. Commenter.

- Tracer l'évolution du gap de dualité en fonction des itérations. Commenter.

- Tracer l'évolution de l'erreur quadratique moyenne entre le signal original $\overline{x}$ et le signal estimé $\widehat{x}_\lambda$ en fonction de λ. Commenter.

2 Problème de régression logistique parcimonieuse

≪ Prenons l'exemple d'une maladie à composante génétique dont on cherche à trouver les gènes impliqués. Il est devenu tellement facile et bon marché d'accumuler les données sur un sujet que l'on mesure tout ce qu'on peut. Au final, le signal qui nous intéresse, c'est-à-dire les variables qui ont quelque chose à voir avec la maladie, est noyé dans un océan de variables qui ne nous intéresse pas. On se retrouve avec des modèles comprenant des centaines de millions de variables dont on sait bien que seul une petite partie a un rapport avec le phénomène que l'on recherche. Du coup les techniques d'optimisation convexe (minimisation de la norme L_1) peuvent s'appliquer et on arrive à résoudre effectivement le système – par exemple à trouver les sites de génome impliqués dans la maladie – en supposant que la solution est parcimonieuse. ≫

– Emmanuel Candès, La Recherche, Fév. 2014 –

Dans ce second exercice nous nous intéressons au problème de sélection de variables dans les problèmes de classification. Plus précisemment, on dispose d'une base d'apprentissage composée de N patients. Parmi ces N patients, un ensemble $S \subset \{1, \dots, N\}$ appartient à la classe des ≪ patients sains ≫ et le restant correspond à la classe des ≪ patients malades ≫ ($M \subset \{1, \dots, N\}$). On note $b = (b_i)_{1 \leq i \leq N}$ un vecteur permettant de qualifier si un patient est sain (*resp.* malade), i.e., pour tout $i \in S$, $b_i \equiv 1$ (*resp.*, pour tout $i \in M$, $b_i \equiv -1$). Pour chaque patient, on dispose d'un ensemble d'informations stocké dans un vecteur $y_i = (y_{i,\ell})_{1 \leq \ell \leq K} \in \mathbb{R}^K$. Les N échantillons fournis pour la phase d'apprentissage peuvent être présentés comme un ensemble $\mathcal{D} = \{(b_1, y_1), \dots, (b_N, y_N)\}$.

On s'intéresse ici au problème d'estimation de la matrice de poids $\widehat{x}_\lambda \in \mathbb{R}^K$ telle que :

$$\widehat{x}_\lambda \in \operatorname*{Argmin}_{x \in \mathbb{R}^K} \sum_{i=1}^{N} \log(1 + \exp(-b_i x^\top y_i) + \lambda \|x\|_1$$

où $\lambda > 0$.

1) Discuter de l'influence du paramètre λ sur la solution $\widehat{x}_\lambda$.

2) Dans MATLAB, télécharger le vecteur b et la matrice y à partir du fichier data_exo2_training.mat

3) Quel est le nombre de patients dans la base d'apprentissage. Combien sont sains (resp. malades) ?

4) Implémenter l'algorithme explicite-implicte pour estimer $\widehat{x}_\lambda$. Tracer l'évolution de la fonction objectif en fonction des itérations. Vérifier la validité de l'estimation de $\widehat{x}_\lambda$ en calculant la quantité
$$\widehat{b}_\lambda = \operatorname{sign}(\widehat{x}_\lambda^\top y)$$
pour $\lambda = \{0.1, 1, 10\}$. Commenter les résultats obtenus.

5) Evaluer les performance de votre estimateur sur la base de données data_exo2_estimation.mat.